

E-Build 基础 API 用户手册

目录

前言

本手册是一本详细介绍 E-BuildAPI 使用的文档。本书主要适用使用 IBE 开发的代理人、航空公司、机场等客户。

本手册按照 IBE 开发包的结构，分门别类的给出了 IBE 每个类与 eTerm 指令的对应关系，对应实现，注意事项，错误信息参考等等。对于开发人员使用 IBE 开发有一定帮助，尤其适用于对 eTerm 使用有一定了解的开发人员。

本手册由林乐健编写第五章、徐正生编写第三、九章、李明编写第二，四章、胡亚平编写第六章、姚一编写第七、八章，由李明统稿。在编写的过程中还得到了其他同事的大力支持，在此表示感谢。

2009 年 7 月

1. 综述

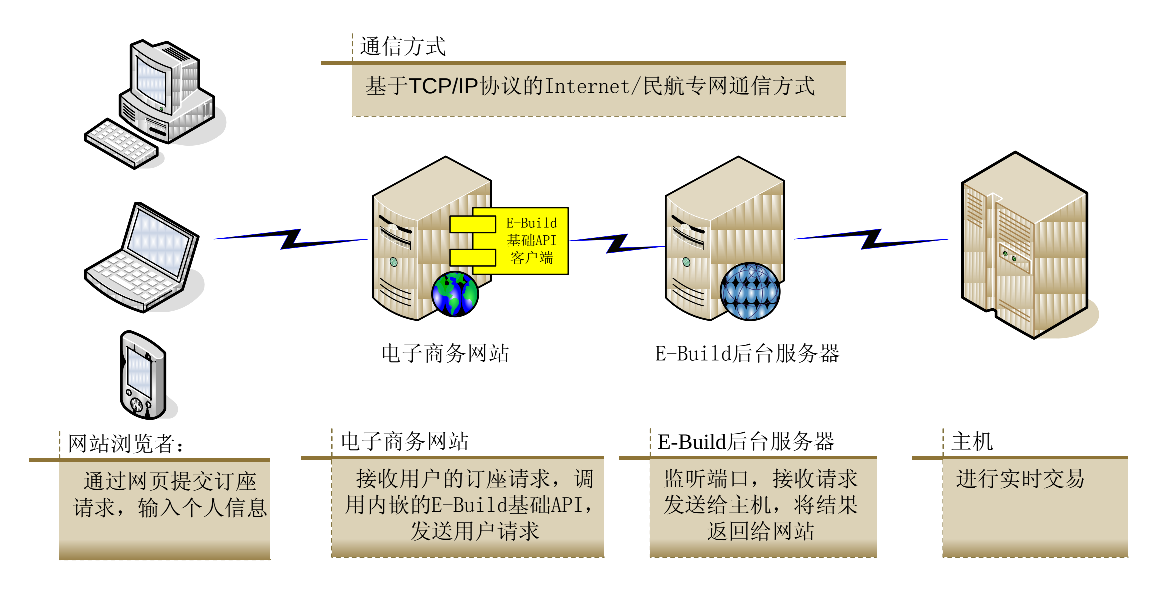
E-Build 基础 API 接口是中国航信提供的电子商务网站解决方案中的底层基础服务，提供了可以直接连接到航信基础平台上各个核心交易系统的程序接口。

E-Build 基础 API 接口功能定位是提供标准的网站访问航信后台基础平台的通道，其本身不涉及到具体的业务逻辑和商务逻辑，功能覆盖航信后台基础平台的高性能访问接口。

E-Build 基础 API 接口采用了标准的 Client/Server 架构模式，Client/Server 之间通过 Socket 协议进行通信。Client/Server 之间的网络环境可以是 Internet 公网或者是中国航信的民航专网。

E-Build 基础 API 客户端是部署在用户电子商务网站上，负责组织网站的数据请求/结果，以及和后台 Server 进行通信的模块。它是用来沟通电子商务网站程序和 E-Build 后台服务器的中间组建。

一个典型的基于 E-Build 基础 API 接口的网站应用架构如下：



1. 用户根据自身的技术开发工具选择适当的 E-Build 基础 API 客户端（Java 或者 COM）
2. 用户申请访问授权，获得访问 E-Build 后台服务器的资质
3. 用户使用 E-Build 基础 API 客户端来构建电子商务网站，现实航班查询，旅客订座
4. 电子商务网站通过调用 E-Build 基础 API 客户端，将请求发送给 E-Build 后台服务器
5. E-Build 后台服务器实时访问航信订座主机，执行指令，返回结果
6. 电子商务网站将 E-Build 基础 API 客户端的返回处理后，展现给用户

2. 航班信息查询

2.1. 的座位可利用情况显示 AV

实现类：com.travelsky.ibe.client.AV
　　　　com.travelsky.ibe.client.AvResult
　　　　com.travelsky.ibe.client.AvItem
　　　　com.travelsky.ibe.client.AvSegment

- eTerm AV 查询过程

eTerm 中运用主机指令查询舱位可利用情况如下：

```
AV:PEKSHA/19JUN09/CA
 19JUN(FRI) BJSSHA VIA CA
1-  CA984   PEKPVG 0720   0930   321 0^S  E  AS#FA A6 P4 O2 YA BA MA HA KA LA*
2   CA1831  PEKSHA 0730   0940   772 0^S  E   DS# FA AS O2 YA BA MA HA KA LA QA*
3  *CA3110  PEKSHA 0815   1005   767 0^S  E   DS# FA YA BA HA KA LA QA GA UA
4   CA1501  PEKSHA 0830   1040   772 0^S  E   DS# FA AS OS YA BA MA HA KA LA QA*
5   CA1519  PEKSHA 0930   1135   777 0^S  E   DS# FA AS OS CA DS ZS IS RS YA BS*
6   CA1995  PEKSHA 1030   1240   772 0^S  E   DS# FA AS O2 YA BA MA HA KA LA QS*
7   CA1557  PEKSHA 1130   1340   772 0^S  E   DS# FA AS OS YA BA MA HA KA LA QS*
8+ *CA3108  PEKSHA 1140   1350   767 0^L  E   DS# FA YA BA HA KA LA QS GA US
```

查询舱位可利用情况(AV)：IBE 可对应 AV 类的 getAvailability(String org,
String dst,
Date date,
String airline,
boolean direct,
boolean e_ticket)
方法。

org：出发城市三字码，如上图：PEK
dst：到达城市三字码，如上图：SHA
date：查询日期，如上图得日期“19JUN09”
airline：指定航空公司，如上图：CA
direct：指明是否直飞航班
e_ticket：是否只查询 et 航班.

返回结果为 AVResult。

- IBE 实现舱位可利用情况查询

```
/*
* 实现舱位可利用情况查询
* */

    /*生成舱位查询对象*/
  AV avExample=new AV();

  /*查询上海至北京国航航班舱位可利用情况*/
AvResult avres= avExample.getAvailability("PEK", "SHA", "20081226 00:00:00","CA",true,true);

  /*显示查询运价结果*/
  System.out.println(avres);

  /*显示指定航班指定日期的查询运价结果*/
  System.out.println("指定航班可利用座位信息");

  System.out.println(avExample.getAvailByFltNo("CA1501","20090619 00:00:00"));
```

AV 查询显示结果

主信息：

查询日期　　始发地　　终到地

19JUN09(FRI)　　PEK　　SHA

查询日期 date:Fri Jun 19 00:00:00 CST 2009

系统信息：

*** NOTE:NO SYSTEM MESSAGES ***

航段信息:

线路 1:

航段 1:

航班号　起飞地　到达地　起飞时间(DATE　　)　到达时间(DATE　　)　ASR　餐食　电子票　经停　代码共享(实际执行航班)　联结协议　机型　登机 TERM　出港 TERM

CA984　PEK　PVG　0720　(Fri Jun 19 07:20:00 CST 2009)　0930　(Fri Jun 19 09:30:00 CST 2009)　true　true true　0　false(　)　321　T3　--

舱位信息

FA

A6

O2

YA

BA

MA

HA

KA

LA

QA

GQ

SA

XQ

NQ

VQ

UQ

WA

TQ

EQ

舱位信息 2

A6 BA EQ FA GQ HA KA LA MA NQ O2 QA SA TQ UQ VQ WA XQ YA

线路 2:

省略...

线路 3:

省略...

```
.
.
.
线路23:
    航段1:
        航班号    起飞地  到达地  起飞时间(DATE                    )      到达时间(DATE                    )  ASR    餐食  电子票  经停  代码共享(实际执行航班)  联结协议  机型    登机TERM    出港TERM
    CA3116    PEK    SHA    2205    (Fri Jun 19 22:05:00 CST 2009)    0010+1(Sat Jun 20 00:10:00 CST 2009)    true    true true    0    true(null)            767    T3        --
                    舱位信息
    FA
    YA
    BA
    HA
    KA
    LA
    QS
    GS
    US

                    舱位信息2
                    BA FA GS HA KA LA QS US YA
```

TIPS：

> IBE 查询航班可利用座位时，一次查询就把所有的可利用座位信息取出来，不需执行多次查询。

出错信息提示:

AVNoRoutingException　　系统暂时没有提供符合查询条件的航班，请查询航空公司文件。

2.2. 航班时刻显示 SK

实现类：com.travelsky.ibe.client.SK
com.travelsky.ibe.client.SkResult
com.travelsky.ibe.client.SkItem
com.travelsky.ibe.client.SkSegment

➢ eTerm SK 查询过程

eTerm 中运用主机指令查询航班时刻如下：

```
SK:PEKSHA/20AUG/CA      --①
17AUG(MON)/23AUG(SUN)  PEKSHA VIA CA
1   CA984   PEKPVG 0720   0930   JET 0 S  E   347  29JUL30AUG FAPOYBMHKL
2   CA984   PEKPVG 0720   0930   321 0 S  E   X347  01AUG31AUG FAPOYBMHKL
3   CA155   PEKPVG 0730   0935   738 0 S  E   4    13AUG20AUG CDZIRJYBMH
4   CA155   PEKPVG 0730   0935   738 0 S  E   367  01AUG24OCT CDZIRJYBMH
5   CA1831  PEKSHA 0730   0940   JET 0 S  E   4    23JUL20AUG FAOYBMHKLQ
6   CA1831  PEKSHA 0730   0940   772 0 S  E        21AUG28AUG FAOYBMHKLQ
7   CA1831  PEKSHA 0730   0940   772 0 S  E   27   11AUG18AUG FAOYBMHKLQ
8+  CA1831  PEKSHA 0730   0940   777 0 S  E   X247 10AUG19AUG FAOCDZIRYB
PAGE 1/1
```

航班时刻显示(SK)：IBE 可对应类 SK 的 getSchedule(String org, String dst, Date date, String airline, boolean direct, boolean nostop)方法。

org：出发城市三字码，如上图：PEK
dst：到达城市三字码，如上图：SHA
date：查询日期，如上图：20AUG
airline：指定航空公司，如上图：CA
direct：指明是否直飞航班
passType：指明是否只查询无经停航班

返回结果为 SKResult

➢ IBE 实现航班时刻显示查询

```
/*
 * 实现航班时刻查询
 * */
/*生成航班时刻查询对象*/
    SK skExample = new SK();

     /*查询时间，为 DATE 类型*/
    Date skDate=QDateTime.stringToDate("20090820","yyyymmdd");

    /*查询北京至上海国航的航班时刻*/
  SkResult skres=skExample.getSchedule("pek","sha",skDate,"ca",true,true);

    /*显示查结果*/
    System.out.println(skres);;
```

SK 查询显示结果

```
查询周起始日   :Mon Aug 17 00:00:00 CST 2009
查询周结束日   :Sun Aug 23 23:59:59 CST 2009
系统讯息:
起飞地:PEK     目的地 SHA

  线路 1:
      航段 1:
        航班号    起飞到达    起飞时间    到达时间    机型 经停次数 OPENTO 连接协议 ASR  餐
食   电子票 代码共享   运行周期    生效日                                  过期日
舱位列表
        CA984    PEKPVG    0720     0930     JET    0                  false  false
false false    34   7     2009 07 29 00:00:00   WED     2009 08 30 00:00:00   SUN     SXNVUWTE
  线路 2:
      省略...
  线路 3:
      省略...
  ……

  线路 52:
      航段 1:
        航班号    起飞到达    起飞时间    到达时间    机型 经停次数 OPENTO 连接协议 ASR  餐
食   电子票 代码共享   运行周期    生效日                                  过期日
舱位列表
        CA1589   PEKSHA    2030     2240     330    0                  false true true
false    1          2009 08 17 00:00:00   MON     2009 08 24 00:00:00   MON     SXNVUWTE
```

出错信息提示:

SKNoRoutingException	制定时间和城市对没有航班设定
SKDateException	查询时选择的日期不正确
SKCityPairException	查询时选择的城市对不正确

2.3. 航班经停点及起降时间的显示 FF

实现类：com.travelsky.ibe.client.FF
com.travelsky.ibe.client.FFResult
com.travelsky.ibe.client.FFSegment

➢ eTerm FF 查询过程

eTerm 中运用主机指令查询航班经停点及起降时间如下：

```
FF:MU2515/20AUG       --①
YIH           1430     EMB
WUH    1500    1830     733
PVG    2000
```

查询航班经停点及起降时间(FF)：IBE 可对应类 FF 的 flightTime(String airNo, String date)方法。

airNo：航班号，如上图：MU2515

date：起飞日期，如上图：20AUG 格式 DDMONYY 或 DDMON

返回结果为 FFResult

➢ IBE 实现航班经停点及起降时间查询

```
/*
 * 实现航班经停点及起降时间查询
 * */
/*生成查询对象*/
   FF   ffExample=new FF();

   /*查询航班 CA929 经停点及起降时间*/
FFResult ffres = ffExample. flightTime ("ca929","20jun09");

   /*显示查询结果*/
   System.out.println(ffres);
```

FF 查询显示结果

航班号：MU2515　　日期：Thu Aug 20 00:00:00 CST 2009　　机型：JET
始发城市：YIH　　到达城市：WUH　　起飞时间：Thu Aug 20 14:30:00 CST 2009　　到达时间：Thu Aug 20 15:00:00 CST 2009　　机型: EMB

始发城市：WUH　　到达城市：PVG　　起飞时间：Thu Aug 20 18:30:00 CST 2009　　到达时间：Thu Aug 20 20:00:00 CST 2009　　机型: 733

。

2.4. 显示航班飞行时间 DSG

实现类：com.travelsky.ibe.client.DSG
　　　com.travelsky.ibe.client.DsgResult
　　　com.travelsky.ibe.client.DsgSegment

➢ eTerm DSG 查询过程

eTerm 中运用主机指令查询航班飞行时间如下：

```
DSG:C/CA981/Y/20AUG      --①
  CA981   Y (THU)20AUG       PEK      1305    744 BC  0  0
                     1420   JFK ELAPSED TIME 14:15 DIST 0
```

显示航班起飞时间(DSG)：IBE 可对应类 DSG 的 displaySegment (String flight, char cabin,String date, String segment)方法。

flight：要查询的航班号，如上图：CA981
cabin：舱位，如上图：Y
date：日期，如上图：20AUG 格式为"DDMMM","DDMMMYY"
segment：航段城市对,可以为空(null)

返回结果为 DSGResult

➢ IBE 实现显示航班飞行时间

```
/*
 * 实现航班飞行时间查询
 * */
/*生成 DSG 对象*/
  DSG   dsgExample=new DSG();

  /*查询成人上海至北京国航运价*/
DSGResult dsgres = dsgExample.findPrice("ca981",'y',"20aug09",null);

  /*显示查询结果*/
  System.out.println(dsgres);
```

DSG 查询显示结果

```
CA981 Y
PEK     20AUG 1305     JFK     20AUG 1420     744   餐食：B
ELAPSE TIME: 14hours15minutes      DISTANCE: 0M
```

TIPS：

IBE DSG 查询航班起飞时间时，只有指定舱位，才能查询出餐食信息
对于结果显示的距离数据由航空公司确定，如有疑义请咨询航空公司。

出错信息提示:

DSGCityPairException	查询的航段不正确。
DSGClassException	给出了错误的舱位。
DSGFlightNumberException	查询的航班号不正确
DSGNoOpException	查询的航班在查询的日期不执行
DSGOptionException	指令使用的查询选项不正确

2.5. 票价查询 FD

实现类：com.travelsky.ibe.client.FD

com.travelsky.ibe.client.FDResult

com.travelsky.ibe.client.FDItem

com.travelsky.ibe.client.Fares

➢ eTerm FD 查询过程

eTerm 中运用主机指令查询票价如下：

```
FD:PEKSHA/19JUN/CA      --①
FD:PEKSHA/19JUN09/CA                /CNY /TPM  1178/
01 CA/F       /  1700.00=  3400.00/F/F/  /   .  /01JUL07       /7001
02 CA/C       /  1470.00=  2940.00/C/C/  /   .  /01JUL07       /7001
03 CA/Y       /  1130.00=  2260.00/Y/Y/  /   .  /01JUL07       /7001
04 CA/B       /  1040.00=  2080.00/B/Y/  /   .  /20APR09       /7001
05 CA/M       /   990.00=  1980.00/M/Y/  /   .  /20APR09       /7001
06 CA/H       /   950.00=  1900.00/H/Y/  /   .  /20APR09       /7001
07 CA/K       /   900.00=  1800.00/K/Y/  /   .  /20APR09       /7001
08 CA/L       /   860.00=  1720.00/L/Y/  /   .  /20APR09       /7001
09 CA/Q       /   770.00=  1540.00/Q/Y/  /   .  /20APR09       /7001
10 CA/G       /   680.00=  1360.00/G/Y/  /   .  /20APR09       /7001
11 CA/V       /   630.00=  1260.00/V/Y/  /   .  /20APR09       /7001
12 CA/U       /   590.00=  1180.00/U/Y/  /   .  /20APR09       /7001
PAGE 1/1
```

① 查询舱位价格(FD)：IBE 可对应类 FD 的 findPrice(String org, String dst, String date, String airline, String planeModel, String passType, boolean fullFareBasis)方法。

org：出发城市三字码，如上图：PEK

dst：到达城市三字码，如上图：SHA

date：查询日期，如上图：19JUN 格式为"DDMMM","DDMMMYY"

airline：指定航空公司，如上图：CA

planeModel：机型（此参数为 null 或者""时表示不查询机场建设费）

passType：旅客类型（"ad"：成人；"IN"：婴儿；"CH"：儿童）

fullFareBasis：应当设置为真

返回结果为 FDResult

➢ IBE 实现舱位运价查询

```
/*
 * 实现舱位运价查询
 * */
/*生成运价查询对象*/
  FD fdExample=new FD();

  /*查询成人上海至北京国航运价*/
FDResult fdres = fdExample.findPrice("pek","sha","13jun09","ca","330","AD",true);

  /*显示查询运价结果*/
  System.out.println(fdres);

  /*查询成人机场建设费*/
  System.out.println("成人机场费："+fdres.getAirportTax(0));
  /*查询成人燃油附加费*/
  System.out.println("成人燃油税："+fdres.getFuelTax(0));

  /*查询儿童机建燃油费*/
   fdres = fdExample.findPrice("pek","sha","13jun09","ca","330","AD",true);
  /*查询儿童机场建设费*/
  System.out.println("儿童机场费："+fdres.getAirportTax(0));
  /*查询儿童燃油附加费*/
  System.out.println("儿童燃油税："+fdres.getFuelTax(0));


  /*查询婴儿机建燃油费*/
   fdres = fdExample.findPrice("pek","sha","13jun09","ca","330","IN",true);
  /*查询婴儿机场建设费*/
  System.out.println("婴儿机场费："+fdres.getAirportTax(0));
  /*查询婴儿燃油附加费*/
  System.out.println("婴儿燃油税："+fdres.getFuelTax(0));
```

FD 查询显示结果

始发地:pek　　到达地:sha　　查询航空公司:--

F	F	F	150.0	1700.00	3400.00	01JUL07	31DEC49	CNY	CA	7001	50.0 0.0
C	C	C	130.0	1470.00	2940.00	01JUL07	31DEC49	CNY	CA	7001	50.0 0.0
Y	Y	Y	100.0	1130.00	2260.00	01JUL07	31DEC49	CNY	CA	7001	50.0 0.0
B	B	Y	92.0	1040.00	2080.00	20APR09	31DEC49	CNY	CA	7001	50.0 0.0
M	M	Y	88.0	990.00	1980.00	20APR09	31DEC49	CNY	CA	7001	50.0 0.0
H	H	Y	84.0	950.00	1900.00	20APR09	31DEC49	CNY	CA	7001	50.0 0.0
K	K	Y	80.0	900.00	1800.00	20APR09	31DEC49	CNY	CA	7001	50.0 0.0
L	L	Y	76.0	860.00	1720.00	20APR09	31DEC49	CNY	CA	7001	50.0 0.0
Q	Q	Y	68.0	770.00	1540.00	20APR09	31DEC49	CNY	CA	7001	50.0 0.0
G	G	Y	60.0	680.00	1360.00	20APR09	31DEC49	CNY	CA	7001	50.0 0.0
V	V	Y	56.0	630.00	1260.00	20APR09	31DEC49	CNY	CA	7001	50.0 0.0
U	U	Y	52.0	590.00	1180.00	20APR09	31DEC49	CNY	CA	7001	50.0 0.0

成人机场费：50.0
成人燃油税：0.0
儿童机场费：50.0
儿童燃油税：0.0
婴儿机场费：0.0
婴儿燃油税：0.0

TIPS：

IBE 查询运价时，不论输入的旅客类型参数是成人(AD)儿童(CH)婴儿(IN)，得出的价格都为成人运价，儿童运价一般为 Y 舱价格的一半，婴儿为 Y 舱价格的十分之一。具体情况请参照各个航空公司的客规。
如果 AV 查询设置航空公司为"ALL",则最好 FD 查询设置航空公司也为"ALL"，这样可以减少交易量，减少您的费用。

出错信息提示:

FDComputeFareManullyException　　系统暂时没有提供符合查询条件的运价，请查询航空公司文件。

2.6. 提取旅客名单的多种选择法 ML

实现类：com.travelsky.ibe.client.ML
com.travelsky.ibe.client.MultiResult
com.travelsky.ibe.client.MultiItem
com.travelsky.ibe.client.MultiSegment

➢ eTerm ML 查询过程

eTerm 中运用主机指令查询指定航班日期旅客名单如下：

```
ML:C/CA1501/20AUG      --①
MULTI
CA1501  /20AUG          B
PEKSHA
 001   1LEWIS/JOSEPH A   BL0F8 F HK2   MUC1A  10JUL09     OQST C
 002   1LEWIS/SOFI      BL0F8 F HK2   MUC1A  10JUL09     OQST C
 003   1LIU/NORMAN      DK4JH F HK1   HDQAA  23JUL09     OQST C
 004   1STASZAK/JEFFREY  Z8XDF F HK1   HDQAA  25JUN09     O ST C
 005   1KITPREECHAVANIC+ B2JZ1 S HK1   MUC1A  06JUN09  I  O ST C
 006   1MARKMAN/MARTHE + C2H3V S HK2   HDQAA  04MAR09  I  O ST C
 007G 41RGN             DS19H S HK41  RGN001 13MAY09       T C
 008   1THORSEN/CHRISTI+ C2H3V S HK2   HDQAA  04MAR09  I  O ST C
 009   1WANGQINGHONG     G92YH L HK1   PEK1E  11MAY09     O S
```

提取旅客名单(ML)：IBE 可对应类 ML 的 multi(String criteriacode, boolean pnrOnce, String flightNo, char fltclass, String date, String city) 方法。

criteriacode：指令参数:可以为空,默认'C',如上图：C
某些选项可以组合,详见 IBEJAVA DOC 文档。

pnrOnce：指令参数:为true表示每一个PNRNO只显示一次,默认为 false

flightNo ：航班号，如上图：CA1501

fltclass：提取舱位选择--默认为' ',即提取所有舱位

date：查询航班的执行日期，如上图：20AUG，格式为"DDMMM","DDMMMYY"

city：起飞城市或起飞到达城市对，可以为空，此时在该航班无经停情况下查询结果为本 office 在该航班上的订座记录

返回结果为 MLResult

➢ IBE 实现旅客名单提取

```
/*
 * 实现指定日期航班旅客姓名提取
 * */
/*生成 ML 对象*/
  ML mlExample=new ML();

  /*查询 09 年 8 月 20 日 CA1501 航班上的旅客名单*/
MLResult mlres = mlExample. multi ("c",false," ca1501",' ',"20AUG",null);

  /*显示查询运价结果*/
  System.out.println(mlres);
```

ML 查询显示结果

```
航班号  : CA1501
日期 : Thu Aug 20 00:00:00 CST 2009
查询参数:      C
-------------------------------------------------------
PEK SHA
001   1   LEWIS/JOSEPH A        BL0F8 F HK2    MUC1A    10JUL09       OQST
002   1   LEWIS/SOFI            BL0F8 F HK2    MUC1A    10JUL09       OQST
003   1   LIU/NORMAN            DK4JH F HK1    HDQAA    23JUL09       OQST
004   1   STASZAK/JEFFREY       Z8XDF F HK1    HDQAA    25JUN09       O ST
....
024N 1    LI/LIU                C19WW Y HX12   PEK1E    23MAR09       O S
....
037   0   ZHANGZHANMIN          BB2DD Y HX1    PEK1E    22JUL09          T
-------------------------------------------------------
TOTAL NUMBER 84
```

出错信息提示:

MLAuthorityException 无权进行 ML 查询的航班。
MLCityPairException 城市对输入不正确。
MLCodeException ML 指令使用的查询选项不正确。
MLDateException ML 查询的日期不正确。
MLFltNumberException ML 查询的航班不正确。
MLNONStoredException 查询航班信息在主机中不存在。

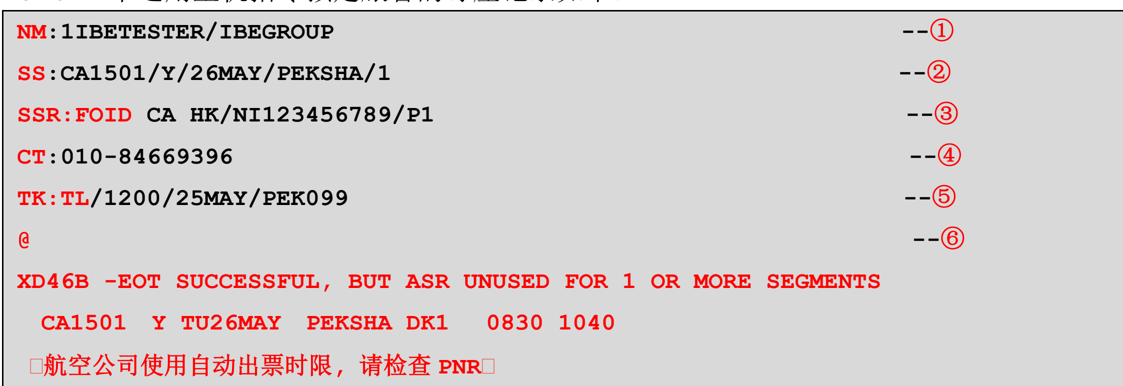
3. 旅客订座记录

实现类：com.travelsky.ibe.client.pnr.SellSeat

3.1. 预定成人单人单程 PNR

➢ eTerm 预定 PNR 过程

eTerm 中运用主机指令预定旅客的订座记录如下：



```
NM:1IBETESTER/IBEGROUP                                  --①
SS:CA1501/Y/26MAY/PEKSHA/1                              --②
SSR:FOID CA HK/NI123456789/P1                           --③
CT:010-84669396                                         --④
TK:TL/1200/25MAY/PEK099                                 --⑤
@                                                       --⑥
XD46B -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA1501  Y TU26MAY  PEKSHA DK1   0830 1040
 □航空公司使用自动出票时限，请检查 PNR□
```

相对于 eTerm 中预定旅客订座记录的主机指令，如 NM、SS、CT、TKTL 等。IBE 把所有与预定旅客订座记录的主机指令的打包在一块，通过 com.travelsky.ibe.client.pnr.SellSeat 类的各个方法实现订座中的主机指令功能。一般，生成一个 PNR，需要输入姓名、航段、联系方式和出票时限四个基本要素。由于现在民航安全的要求，在订票时，一般都要求输入真实身份信息，因此上面的例子除了四个基本要素之外添加一个身份信息生成订座记录(pnr)。常见的 PNR 可能还会有其他的 SSR，OSI，RMK 需要输入，可以从后面的章节中看到在 IBE 中对应的方法。

① 输入旅客姓名(NM)：可对应 addAdult(String name)方法。

name：要添加的成人旅客姓名，似上图：IBETESTER/IBEGROUP

如果需要添加多个成人旅客，可多次调用此方法，并为每一格旅客输入身份信息，需要注意的是，虽然主机支持，但是通过 IBE 生成 PNR 时，不要添加多个完全同名的旅客，如果有这种情况的需求，可以生成多个 PNR 来保证没有同名旅客

② 预定航段组(SS)：可对应 addAirSeg(String airNo, char fltClass, String orgCity, String desCity, String actionCode, int tktNum, String departureTime)方法。

airNo：航班号，似上图：CA1501
fltClass:舱位等级，似上图：Y
orgCity：出发城市三字码，似上图：PEK
desCity：到达城市三字码，似上图：SHA

actionCode：行动代码，一般设为“NN”，如果是需要候补，可用”LL”
tktNum：订座人数，似上图：1
departureTime：出发时间，似上图 26MAY

③ 输入旅客身份信息(SSR:FOID)：可对应 addSSR_FOID(String airline, String idtype, String id, String name)方法。
airline：航空公司代码，似上图：CA
idtype：旅客证件信息，似上图：NI
id：证件号码：

④ 输入联系组(CT)：可对应 addContact(String contactinfo)方法。
contactinfo 为预定 PNR 旅客的联系信息

⑤ 出票时限(TK:TL)：可对应 setTimelimit(String dateLimit)方法
dateLimit 出票时限，如上图的 5 月 25 日 1200 点

⑥ 提交(@) 可对应 commit1()方法。

产生的 PNR 结果：

```
1.IBETESTER/IBEGROUP XD46B
2.  CA1501 Y  TU26MAY  PEKSHA HK1   0830 1040        E T3--
3.010-84669396
4.TL/1200/25MAY/PEK099
5.SSR FOID CA HK1 NI123456789/P1
6.RMK CA/TD54T
7.PEK099
```

➢ IBE 实现成人单航段 PNR 预定

```
/*
 * 成人单人单航段预定示例
 *
 * */
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加旅客乘坐航段信息*/
String airNo="CA1501";
```

```
char  fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-05-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客身份证信息*/
String airline="CA";
String idtype="NI";
String id="123456789";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);

/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-05-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

TIPS：

输入旅客姓名时：不能输入两个姓名完全一样的旅客
设置 TKTL 时： 保证时间在飞机起飞前。
对于国内航空公司的国内票，如出电子客票，则身份信息中 SSR FOID 中的证件类型都使用“NI”，无论证件是身份证、护照或其他。
IBE 的 SellSeat 在设置提交订座信息时，尽量使用 commit1()方法，此方法会返回航段订取后的状态信息，一般来说，如果是 DK、HK 状态，则占座成功，如果希望生成占座成功的航段，但是返回的全部/部分航段状态是 HN，HL 等的话，则可能需要再次 RT 查看占座是不是获得了确认。
如果是在代理人系统订座，并且订取的航班是国内航空公司航班，系统会在 PNR 中自动生成 RMK CA/XXXXX 行，其中的 XXXXX 表示的是该订座记录在航空公司系统对应的订座记录的编号。

➢ 关于订座提交的结果

eTerm 返回结果形式如下：

```
XD46B -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA1501  Y TU26MAY  PEKSHA DK1   0830 1040
 航空公司使用自动出票时限，请检查 PNR
```

使用 IBE 的 SellSeat 类提交订座信息，使用 commit()方法，订座成功只会返回一个主机给出的 PNR 编号，如上的“XD46B”，IBE 推荐使用 commit1()方法，对于国内航空公司的订座，能将航段信息，尤其是航段订取后的状态信息一次返回给 IBE 调用程序。

方法的定义如下：

public SSResult commit1()

SSResult 提供了以下方法：

public String getPnrno() 获取 pnr 编号。

public int getSegmentsCount() 获取封口信息里列出的航段数

public SSSegment getSegmentAt(int i) 获取封口信息里列出的第 i 航段，从 0 开始计数。

SSSegment 提供的方法在检查订座封口信息时常用的：

public String getAirNo() 获取航班号

public char getFltClass() 获取舱位

public String getOrgCity() 获取起飞机场三字码

public String getDesCity() 获取到达机场三字码

public String getActionCode() 获取提交主机订座请求后主机返回的状态

一般通过这些方法可以检查是否与向 SellSeat 添加的航段是否一致，包括航班号、舱位等。在航班信息一致的情况下，需要检查 ActionCode 是否为 DK/HK/TK/KK/KL。订取散客票时，如果返回的不是这几个状态之一，则座位并未实际占座，需要等待或者与航空公司联系以确认座位，在确认之前，不要继续一般的支付和出票流程，否则，可能会生成状态不是 OK 的电子客票航段。

对订座结果的展示程序如下：

```
System.out.println("PNR NO is:"+ssr.getPnrno());
for (int i = 0; i < ssr.getSegmentsCount(); i++){
  SSSegment segment = ssr.getSegmentAt(i);

  System.out.print("航段:   航班号:"+segment.getAirNo());
  System.out.print("  舱位:"+segment.getFltClass());
  System.out.print("  从:"+segment.getOrgCity()+"  飞往:"+segment.getDesCity());
  System.out.println("  航段订座状态:"+segment.getActionCode());
  System.out.print("           起飞时间:"+segment.getDepartureTime());
  System.out.println("  到达时间:"+segment.getArrivalTime());

}
```

输出：

```
PNR NO is:XD46B
```

```
航段:  航班号:CA1501 舱位:Y 从:PEK 飞往:SHA 航段订座状态:DK
          起飞时间:Tue May 26 08:30:00 CST 2009 到达时间:Thu May 26 10:40:00 CST 2009
```

3.2. 预订成人往返单程 PNR

➢ eTerm 预定 PNR 过程

eTerm 中运用主机指令预定旅客往返程的订座记录如下：

```
Nm: 1IBETESTER/IBEGROUP
SS:CA1501/Y/26DEC09/PEKSHA/NN1
SS:CA1522/Y/29DEC09/SHAPEK/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/25DEC09/pek099
SSR:FOID CA hk/ NI123456789/p1
@
XD41C -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA1501  Y SA26DEC  PEKSHA DK1   0830 1035
  CA1522  Y TU29DEC  SHAPEK DK1   1750 2020
 □航空公司使用自动出票时限，请检查 PNR□
```

实际上对于订座来说，来回程与单程的区别就是订取两个航段，并且第二航段的出发地和到达地与第一航段的到达地和出发地一致。就是将采集的订座需要的数据再次调用添加航段得方法。

产生的 PNR 结果：

```
 1.IBETESTER/IBEGROUP XD41C
 2.  CA1501 Y   SA26DEC  PEKSHA HK1   0830 1035         E T3--
 3.  CA1522 Y   TU29DEC  SHAPEK HK1   1750 2020         E --T3
 4.010-84669396
 5.PEK/T
 6.TL/1200/25DEC/PEK099
 7.SSR FOID CA HK1 NI123456789/P1
 8.SSR ADTK 1E BY PEK23DEC09/0830 OR CXL CA1501 Y26DEC
 9.RMK CA/ZY4B1
10.PEK099
```

➢ IBE 实现成人往返程 PNR 预定

```
        /*
         * 成人往返程预定示例
```

```
 *
 * */
 /*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加旅客乘坐航段信息*/
String airNo="CA1501";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-12-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客乘坐返程航段信息*/
String airNo2="CA1522";
char    fltClass2='Y';
String orgCity2="SHA";
String dstCity2="PEK";
String actionCode2="NN";
int tktNum2=1;
String departureTime2="2009-12-29";
SellSeatexample.addAirSeg(airNo2,fltClass2,orgCity2,dstCity2,actionCode2,tktNum2,departureTime2);

/*添加旅客身份证信息*/
String airline="CA";
String idtype="NI";
String id="123456789";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);

/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-12-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
```

```
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

实际上，订取更多航段的多段 PNR 也可以类似上面的例子，只要按顺序多次调用 addAirSeg 方法顺序加入航段就可以了。需要注意的是各航段的时间顺序和 qifei 到达地的前后连续。

TIPS:

由于 IBE 并不判断航段的先后，对于多个航段的订座，各航段输入的先后顺序不重要，主机自动根据航段时间和航段连续的要求判断航段先后。因此，需要客户端程序自行判断来回程的日期先后限制，否则，如上例，本需订北京到上海的往返程，如果返程日期靠前，则会生成上海到北京的往返程 PNR。

每位旅客的 SSR FOID 身份信息只需要输入一次，一般使用首航段的航空公司代码输入 SSR 即可。如果输入了多次，会报告异常，信息为 DUP ID。

返回信息中的 **SSR ADTK 1E BY PEK23DEC09/0830 OR CXL CA1501 Y26DEC** 表示航空公司自动给出的出票时限，这个出票时限可以通过 RT 生成的 PNR 操作中读到这个文本信息。因此，PNR 中可能有两个出票时限，自行设置的出票时限（在程序中设置的，如此例中的 Timelimit，2009 年 12 月 25 日 12：00），系统给出的出票时限，如此例中的北京时间 2009 年 12 月 23 日 08：30）。无论有一个或者两个出票时限，当任何一个过期之后，如果此 PNR 仍然没有出票，则已经占座的航段有可能自动被取消。

3.3. 订成人缺口程 PNR

所谓缺口，即两个连续的航段之间首尾不连续，因此需要添加一个航段来保证航程的连续性，这个航段称为信息航段，也称地面运输航段，比如一个行程是北京-上海，杭州-北京，为了保证 pnr 中航段连续，需要添加一个地面运输航段。

- eTerm 预定缺口程 PNR 过程

eTerm 中运用主机指令预定旅客带地面运输航段旅程的订座记录如下：

```
Nm: 1IBETESTER/IBEGROUP
SS:CA1501/Y/26DEC09/PEKSHA/NN1
SA: SHAHGH                                   ①
SS:CA1703/Y/29DEC09/HGHPEK/NN1
CT:010-84669396
CT:pek/T
```

```
TK:TL/1200/25DEC09/pek099
SSR:FOID CA hk/ NI123456789/p1
@
XD41V -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA1501  Y SA26DEC  PEKSHA DK1   0830 1035
  CA1703  Y TU29DEC  HGHPEK DK1   0855 1105
 □航空公司使用自动出票时限，请检查 PNR□
```

① SA：订取信息航段，在订座时使用 SA 指令，相对于 BookAirSeg 的信息航段构造函数 BookAirSeg(String orgCity, String desCity)。这个方法没有向航段输入除了起始和到达机场三字码对以外的任何日期、舱位等信息。然后调用 SellSeat 的 addAirSeg(BookAirSeg airseg)方法将信息航段加入到订座信息中。

SA：订取信息航段，也可以 addInfoAirSeg (String orgCity, String desCity) 方法。两个参数分别为需要添加的信息航段的起始机场代码和到达机场代码。

产生的 PNR 结果：

```
1.IBETESTER/IBEGROUP XD41V
 2.  CA1501 Y   SA26DEC  PEKSHA HK1   0830 1035         E T3--
 3.   ARNK              SHAHGH
 4.  CA1703 Y   TU29DEC  HGHPEK HK1   0855 1105     DACSE
 5.010-84669396
 6.PEK/T
 7.TL/1200/25DEC/PEK099
 8.SSR FOID CA HK1 NI123456789/P1
 9.SSR ADTK 1E BY PEK23DEC09/0830 OR CXL CA1501 Y26DEC
10.RMK CA/ZY4CD
11.PEK099
```

TIPS：

```
XD41V -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA1501  Y SA26DEC  PEKSHA DK1   0830 1035
  CA1703  Y TU29DEC  HGHPEK DK1   0855 1105
 □航空公司使用自动出票时限，请检查 PNR□
```

如例子中展示，在封口信息中并没有出现信息航段。封口后的信息中，只会出现普通航段的信息

一般情况下，PNR 中首航段不应该是信息航段。

- IBE 实现成人缺口程 PNR 预定

```
/*
  * 成人缺口程预定示例
  *
  * */
  /*生成预定类对象*/
 SellSeat SellSeatexample= new SellSeat();
```

```
/*添加旅客姓名*/
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加旅客乘坐航段信息*/
String airNo="CA1501";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-12-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*信息航段的添加*/
BookAirSeg arnkSeg;
arnkSeg = new BookAirSeg("SHA","HGH");
SellSeatexample.addAirSeg(arnkSeg);

/*添加旅客乘坐返程航段信息*/
String airNo2="CA1703";
char    fltClass2='Y';
String orgCity2="HGH";
String dstCity2="PEK";
String actionCode2="NN";
int tktNum2=1;
String departureTime2="2009-12-29";
SellSeatexample.addAirSeg(airNo2,fltClass2,orgCity2,dstCity2,actionCode2,tktNum2,departureTime2);

/*添加旅客身份证信息*/
String airline="CA";
String idtype="NI";
String id="123456789";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);

/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-12-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
```

```
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

3.4. 订取含不定期航段的 PNR

不定期航段也称 OPEN 航段，只确定了行程和舱位，但没有确定日期。有时，旅客在订票时尚不能确定某一段或多段航程的旅行日期，只是要求先出票确定行程。如果舱位允许销售 OPEN 票，可以生成含不定期航段的 PNR。

- eTerm 预定不定期航段 PNR 过程

eTerm 中运用主机指令预定旅客带 OPEN 航段旅程的订座记录如下：

```
Nm: 1IBETESTER/IBEGROUP
SS:CA1501/Y/26DEC09/PEKSHA/NN1
SNCA/ Y/ SHAPEK                                    ①
CT:010-84669396
CT:pek/T
TK:TL/1200/25DEC09/pek099
SSR:FOID CA hk/ NI123456789/p1
@
XD426 -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA1501  Y SA26DEC  PEKSHA DK1   0830 1035
 航空公司使用自动出票时限, 请检查 PNR
```

① SN：订取 OPEN 航段，在订座时使用 SN 指令，对应 SellSeat 的 addOpenAirSeg(String airNo, char fltClass, String orgCity, String desCity)方法.

airNO:航空公司代码，如“CA”

fltClass:舱位代码，如‘Y’

orgCity:该航段的出发机场/城市三字码，如“SHA”

desCity:该航段的到达机场/城市三字码，如“PEK”

产生的 PNR 结果：

```
1.IBETESTER/IBEGROUP XD426
2.  CA1501 Y   SA26DEC  PEKSHA HK1   0830 1035        E T3--
3.  CAOPEN Y            SHAPEK
4.010-84669396
5.PEK/T
6.TL/1200/25DEC/PEK099
7.SSR FOID CA HK1 NI123456789/P1
8.SSR ADTK 1E BY PEK23DEC09/0830 OR CXL CA1501 Y26DEC
9.RMK CA/ZY4D0
```

```
10.PEK099
```

TIPS:

XD426 -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
CA1501 Y SA26DEC PEKSHA DK1   0830 1035
□航空公司使用自动出票时限，请检查 PNR□

如例子中展示，类似信息航段，在封口信息中并没有出现 OPEN 航段。封口后的信息中，只会出现普通航段的信息

所有航段，包括信息航段，OPEN 航段和普通航段，一般情况下，应当保证依次是首尾相接连续的。

一般情况下，PNR 中首航段不应该是 OPEN 航段。

- IBE 实现成人信息航段 PNR 预定

```
/*
  * 成人信息航段预定示例
  *
  * */
  /*生成预定类对象*/
 SellSeat SellSeatexample= new SellSeat();

 /*添加旅客姓名*/
 String name="IBETESTER/IBEGROUP";
 SellSeatexample.addAdult(name);

 /*添加旅客乘坐航段信息*/
 String airNo="CA1501";
 char    fltClass='Y';
 String orgCity="PEK";
 String dstCity="SHA";
 String actionCode="NN";
 int tktNum=1;
 String departureTime="2009-12-26";
 SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);


 /*添加旅客乘坐返程航段信息—为 OPEN 航段*/
 String airNo2="CA";
 char    fltClass2='Y';
 String orgCity2="SHA";
 String dstCity2="PEK";
 SellSeatexample.addOpenAirSeg(airNo2,fltClass2,orgCity2,dstCity2);
```

```
/*添加旅客身份证信息*/
String airline="CA";
String idtype="NI";
String id="123456789";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);

/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-12-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult   ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

3.5. 订取成人加儿童的 PNR

儿童在订取使与成人没有太大的不同，但是需要注意的是，航空公司一般要求在姓名后加 CHD 标识。在 IBE 中提供了专门的方法在 PNR 中订取儿童旅客。
儿童也需要输入其对应的 SSR FOID 身份信息。对于没有身份证件的儿童，需要向航空公司查询规定应当输入什么作为身份信息。

➢ eTerm 预定含儿童 PNR 过程

eTerm 中运用主机指令预定成人加儿童旅程的订座记录如下：

```
Nm: 1IBETESTER/IBEGROUP 1 儿童旅客 CHD
SS:CA1501/Y/26DEC09/PEKSHA/NN2
CT:010-84669396
CT:pek/T
TK:TL/1200/25DEC09/pek099
SSR:FOID CA hk/ NI123456789/p1
SSR:FOID CA hk/ NI234567890/p2
@
XD42D -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA1501  Y SA26DEC  PEKSHA DK2   0830 1035
```

```
航空公司使用自动出票时限，请检查 PNR
```

① SN：订取儿童旅客，在订座时使用 SN 指令，对应 SellSeat 的 addChild(String name)方法.

name:即该儿童旅客的姓名，请保持与证件上一致。

如有多个儿童，可多次调用此方法并输入每个儿童的身份信息，同样的，不论成人儿童旅客，在一个 PNR 中不应当有姓名完全一致的两个或多个旅客。

产生的 PNR 结果：

```
1.儿童旅客 CHD 2.IBETESTER/IBEGROUP XD42D
 3.  CA1501 Y   SA26DEC  PEKSHA HK2   0830 1035         E T3--
 4.010-84669396
 5.PEK/T
 6.TL/1200/25DEC/PEK099
 7.SSR FOID CA HK1 NI123456789/P1
 8.SSR FOID CA HK1 NI234567890/P2
 9.SSR ADTK 1E BY PEK23DEC09/0830 OR CXL CA1501 Y26DEC
10.RMK CA/ZY4DE
11.PEK099
```

TIPS:

国内航空公司的国内票，当使用 SSR FOID 格式输入儿童的身份信息，无论是输入生日作为身份信息或者是各种证件号，均使用 NI 作为证件类型

➢ IBE 实现成人加儿童 PNR 预定

```
/*
 * 成人加儿童预定示例
 *
 * */
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加旅客姓名*/
String name_c="儿童旅客";
SellSeatexample.addChild(name_c);
```

```
/*添加旅客乘坐航段信息*/
String airNo="CA1501";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=2;
String departureTime="2009-12-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);



/*添加旅客身份证信息*/
String airline="CA";
String idtype="NI";
String id="123456789";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);

/*添加旅客身份证信息*/
String airline_c="CA";
String idtype_c="NI";
String id_c="234567890";
SellSeatexample.addSSR_FOID(airline_c,idtype_c,id_c,name_c);

/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-12-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

3.6. 订取成人带婴儿的 PNR

国内航空公司一个成人旅客只允许携带一个不占座婴儿，超出的必须占座，这种情况可以参照儿童订取。

婴儿在订取时必须输入其出生日期，由于航空公司能力有限，对于婴儿乘机，航空公司在订

取 PNR 时还可能要求为每个婴儿的每个航段输入 SSR INFT 进行申请，如果航空公司要求输入此项则必须在该申请获得航空公司批准，即该 SSR 由申请时的 NN 状态变成 KK 或者 HK 状态，之后才可以为婴儿出票。批准的过程可能是自动的也可能是手动的。因此，可能生成 PNR 后即已状态变更为 KK，也可能需要一段时间或者营业员主动联系后才会有状态变更，如果状态一直为 NN 或者被变更为 NO，请不要为婴儿出票，同行成人是否还需要单独出票，需要再次联系旅客。

➢ eTerm 预定成人带婴儿 PNR 过程

eTerm 中运用主机指令预定成人带婴儿旅程的订座记录如下：

```
Nm: 1IBETESTER/IBEGROUP
SS:CA1501/Y/26DEC09/PEKSHA/NN1
SS:CA1502/Y/29DEC09/SHAPEK/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/25DEC09/pek099
XN:IN/王婴儿 INF(JAN09)/p1                                              ①
SSR:FOID CA hk/ NI123456789/p1
SSR:INFt CA nn1 PEKSHA 1501 Y 26DEC09 WANG/YINGER 03JAN09/p1            ②
SSR:INFt CA nn1 SHAPEK 1502 Y 29DEC09 WANG/YINGER 03JAN09/p1
@
XD42J -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA1501  Y SA26DEC  PEKSHA DK1   0830 1035
  CA1502  Y TU29DEC  SHAPEK DK1   1150 1415
 □航空公司使用自动出票时限, 请检查 PNR□
```

① 输入婴儿信息，对应 addInfant(String birth, String carrierName, String name)方法。

birth:婴儿生日，目前必须是 yyyy-MM-dd 格式

carrierName:同行携带此婴儿的成人旅客姓名

name:婴儿的姓名，应与证件一致，如果是国内客票且证件为中文姓名，应输入中文姓名

② 输入婴儿的 SSR INFT 申请，某些航空公司要求订座时为**每个**婴儿的**每个**已经确定的航段输入 SSR INFT 项进行申请。输入此项可对应方法 addSSR_INFT(String airline, String segment, String fltNo, String fltClass, String date, String infantName, String infantBirth, String freeText,String carrierName) 方法。

airline:该航段承运人的两字代码，如例中的“CA”，

segment:该航段的起始到达地三字代码对，如“PEKSHA”

fltNo:该航班的航班号，不带航空公司代码，如“1501”

fltClass:同行成人旅客的订座舱位，如“Y”

date:旅行日期，即相应航段日期，如“26DEC09”，格式为“DDMMM”/“DDMMMYY”

infantName:婴儿姓名，格式为 XX/XXX，不接受中文姓名．如此例中“WANG/YINGER”，对于中文姓名的婴儿，需要将姓名转换为拼音，且必须使用 XXX/XXXX 样式输入姓名，即姓与名的分隔符“/”不能省略。

infantBirth:婴儿生日，格式“DDMMMYY”,如“03JAN09”

freeText:如有额外需要的注意的，可写在这里，如果没有，设置为空串

carrierName:同行携带此旅客的成人姓名，应当与订座时输入的该旅客姓名完全一致

产生的 PNR 结果:

```
 1.IBETESTER/IBEGROUP XD42J
 2.  CA1501 Y   SA26DEC  PEKSHA HK1   0830 1035         E T3--
 3.  CA1502 Y   TU29DEC  SHAPEK HK1   1150 1415         E --T3
 4.010-84669396
 5.PEK/T
 6.TL/1200/25DEC/PEK099
 7.SSR FOID CA HK1 NI123456789/P1
 8.SSR ADTK 1E BY PEK23DEC09/0830 OR CXL CA1501 Y26DEC
 9.SSR INFT CA □KK1□PEKSHA 1501 Y26DEC WANG/YINGER 03JAN09/P1
10.SSR INFT CA □KK1□SHAPEK 1502 Y29DEC WANG/YINGER 03JAN09/P1
11.OSI YY 1INF WANGYINGER/P1
12.RMK CA/ZY4E2
13.XN/IN/王婴儿 INF(JAN09)/P1
14.PEK099
```

TIPS:

此例中在订完 PNR 号之后可以看到，航段状态没有问题，都是 HK，而且两个 SSR INFT 都被确认为 KK 状态（订取时使用的代码是 NN），这样 PNR 中的成人和婴儿都可以继续后续的支付出票流程。

如果不是所有的 INFT 都是 HK 或者 KK 状态，建议不要做后续支付出票操作，以免婴儿无法成行带来麻烦。

- IBE 实现成人带婴儿 PNR 预定

```
 /*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加婴儿的基本信息*/
String infant_name = "王婴儿";
String infant_birth = "2009-01-03";
SellSeatexample.addInfant(infant_birth,name,infant_name);




/*添加旅客乘坐航段信息*/
```

```
String airNo="CA1501";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-12-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客乘坐航段信息*/
String airNo2="CA1502";
char    fltClass2='Y';
String orgCity2="SHA";
String dstCity2="PEK";
String actionCode2="NN";
int tktNum2=1;
String departureTime2="2009-12-29";
SellSeatexample.addAirSeg(airNo2,fltClass2,orgCity2,dstCity2,actionCode2,tktNum2,departureTime2);



/*添加旅客身份证信息*/
String airline="CA";
String idtype="NI";
String id="123456789";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);

/*为婴儿的第 0 航段(即去程)添加 SSR INFT */
String airline0_i = "CA";
String segment0_i = "PEKSHA";
String fltNo0_i = "1501";
String fltclass0_i = "Y";
String date0_i = "26DEC09";
String infantName = "WANG/YINGER";
String infantBirth = "03JAN09";
String freeText = "";
SellSeatexample.addSSR_INFT(airline0_i, segment0_i, fltNo0_i, fltclass0_i,
date0_i, infantName, infantBirth, freeText, name);

/*为婴儿的第 1 航段(即回程)添加 SSR INFT */
String airline1_i = "CA";
String segment1_i = "SHAPEK";
String fltNo1_i = "1502";
String fltclass1_i = "Y";
```

```
        String date1_i = "29DEC09";
        SellSeatexample.addSSR_INFT(airline1_i, segment1_i, fltNo1_i, fltclass1_i,
date1_i, infantName, infantBirth, freeText, name);

        /*添加旅客联系组信息*/
        String contactinfo="010-84669396";
        SellSeatexample.addContact(contactinfo);

        /*添加旅客出票时限*/
        String dateLimit="2009-12-25 12:00:00";
        SellSeatexample.setTimelimit(dateLimit);

        /*完成 PNR 必需信息输入，递交主机，生成 PNR*/
        SSResult   ssr=SellSeatexample.commit1();

        /*PNR 结果*/
        System.out.println(ssr.getPnrno());
```

3.7 向 PNR 中添加 SSR/RMK/OSI

在订座时，除了上面提到的几种特殊格式的 SSR 外，还有很多 SSR 代码，如餐食申请的 SPML、常旅客 FQTV、轮椅旅客 WCHR，靠窗座位 NSSW，靠走廊座位 NSSB 等，有的如常旅客 FQTV 在 SellSeat 中有专用方法 addSSR_FQTV,而更多的并没有给出特定的方法，需要用 addSSR 方法添加，有的时候还需要向 PNR 中添加 RMK 项，其他服务 OSI 项。

3.7.1 向 PNR 中添加 SSR

SellSeat 提供了方法：public void addSSR(BookSSR ssr)
需要先自行构造一个 BookSSR 类，然后添加到订座信息中。
BookSSR 提供的构造函数如下：
public BookSSR(String serveCode, String airCode, String actionCode, int person, String cityPair, String airNo, char fltClass, String departureTime, String serveInfo, String psgrName)

serveCode：SSR 的类型代码，如 OTHS，FQTV 等

airCode：SSR 申请的航空公司代码，如无需要，则使用空串“”

actionCode：SSR 申请的代码，根据 SSR 的类型，不同类型有不同的需求

person：此 SSR 为几人输入，如果 SSR 类型要求输入为“/”，则使用 0

cityPair：与航段有关的 SSR 类型在这里输入该航段的起飞到达机场对

airNo：与航段有关的 SSR 类型在这里输入该航段的航班号，不加前面的航空公司两字代码

fltClass：与航段有关的 SSR 类型在这里输入该航段的订座舱位

departureTime：与航段有关的 SSR 类型在这里输入该航段的出发日期，格式 ddMMM 或者 ddMMMyy

serveInfo：其他的需要注明的 SSR 申请信息

psgrName：如此 SSR 与某旅客关联，则输入该旅客的姓名，与录入到姓名组中的姓名应当完全一致。

下面给出几个例子：

旅客信息同 3.1 节中的单人 PNR：

以 FQTV，NSSW,SPML 和 OTHS 为例

eTerm 指令：

```
Nm: 1IBETESTER/IBEGROUP
SS:MU5102/Y/26JUN09/PEKSHA/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/25JUN09/pek099
SSR:FOID MU hk/ NI1234567890987654321/p1
SSR:OTHS MU/ THIS IS TEST PNR
SSR:FQTV MU hk/ MU600123456/p1
SSR:NSSW MU nn1 PEKSHA 5102 Y 26JUN09/p1
SSR:SPML MU nn1 PEKSHA 5102 Y 26JUN09 NO SALT MEAL/p1
@
```

输入后 PNR 如

```
1.IBETESTER/IBEGROUP XD487
 2.  MU5102 Y   FR26JUN  PEKSHA HK1   1300 1500         E T2--
 3.010-84669396
 4.PEK/T
 5.TL/1200/25JUN/PEK099
 6.SSR FOID MU HK1 NI1234567890987654321/P1
 7.SSR SPML MU HN1 PEKSHA 5102 Y26JUN NO SALT MEAL/P1              ⑤
 8.SSR OTHS MU THIS IS TEST PNR                                    ③
 9.SSR NSSW MU HN1 PEKSHA 5102 Y26JUN/P1                           ④
10.SSR FQTV MU HK/ MU600123456/P1                                  ①②
11.SSR ADTK 1E BY PEK23JUN09/1300 OR CXL MU5102 Y26JUN
12.RMK CA/ZY4LT
13.PEK099
```

IBE 代码如下：

```
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加旅客乘坐航段信息*/
String airNo="MU5102";
```

```
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-06-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客身份证信息*/
String airline="MU";
String idtype="NI";
String id="123456789098765432";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);

/*添加 SSR OTHS*/                                                                    ③
SellSeatexample.addSSR(new BookSSR("OTHS", airline, "", 0, "", "", ' ', "", "THIS IS TEST PNR", ""));
/*添加 SSR NSSW*/                                                                    ④
SellSeatexample.addSSR(new BookSSR("NSSW", airNo.substring(0,2),
        "nn", 1, orgCity+dstCity, airNo.substring(2), fltClass, departureTime, "", name));
/*添加 SSR SPML*/                                                                    ⑤
SellSeatexample.addSSR(new BookSSR("SPML", airNo.substring(0,2),
        "nn", 1, orgCity+dstCity, airNo.substring(2), fltClass, departureTime, "NO SALT MEAL", name));


/*常旅客信息*/
String ffp_card_type = "MU" ;                                                  ①
String ffp_card_no = "600123456";
SellSeatexample.addSSR_FQTV(airline,ffp_card_type+ffp_card_no,name);
//SellSeatexample.addSSR(new BookSSR("FQTV", airline,                          ②
            "hk", 0, "", "", ' ', "", ffp_card_type+ffp_card_no, name));


/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-06-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
```

```
System.out.println(ssr.getPnrno());
```

代码中的

① SellSeatexample.addSSR_FQTV(airline,ffp_card_type+ffp_card_no,name);

② 或者 SellSeatexample.addSSR(new BookSSR("FQTV", airline,
"hk", 0, "", "", ' ', "", ffp_card_type+ffp_card_no, name))；

是等价的，生成 PNR 后对应得 SSR FQTV 的格式均是

SSR FQTV MU HK/ MU600123456/P1

第一种形式的代码对应为 FQTV 输入的专用方法：

addSSR_FQTV(String airline, String cardno, String name)

airline 参数为执行航班的航空公司代码，例中航班为 MU5101，因此赋值为 MU

cardno 为卡号，包括常客卡所属航空公司代码和常客卡编号，例中举例为东航东方万里行卡，编号为 600123456。实际订座中，该卡号的持卡人姓名应与订座时旅客姓名一致。

name 为持卡人姓名，与订座时的旅客姓名一致。

第二种形式的代码为 SSR 的通用输入方法，在构造 BookSSR 时，使用构造函数 BookSSR(String serveCode, String airCode, String actionCode, int person, String cityPair, String airNo, char fltClass, String departureTime, String serveInfo, String psgrName)

serveCode:因为是常旅客信息输入，代码为 FQTV;

airCode:执行航班的航空公司，为 MU

actionCode:行动代码，按 FQTV 的输入规范，为 HK

person:FQTV 这种格式的输入规范，应该写入“/”，因此此参数为 0

cityPair:FQTV 这种格式的输入规范无需输入，因此赋值为空串

airNo: FQTV 这种格式的输入规范无需输入，因此赋值为空串

fltClass: FQTV 这种格式的输入规范无需输入，因此赋值为空格字符

departureTime: FQTV 这种格式的输入规范无需输入，因此赋值为空串

serveInfo: FQTV 这种格式的输入规范为含航空公司代码的卡号，因此赋值为卡所属航空公司加卡号

psgrName: 持卡人姓名，与订座时旅客姓名一致。体现在 PNR 中表现为该条 SSR 以“/Pn”形式结尾，如例中的“/P1”

③添加 SSR OTHS 时，OTHS 类型的 SSR 用于说明一些告知或注明的信息，IBE 也是使用标准 SSR 添加格式向订座信息中添加 SSR OTHS，生成后在 PNR 中显示为：

SSR OTHS MU THIS IS TEST PNR

则 IBE 代码为

SellSeatexample.addSSR(new BookSSR("OTHS", airline, "", 0, "", "", ' ', "", "THIS IS TEST PNR", ""));

serveCode:因为是常旅客信息输入，代码为 OTHS;

airCode:执行航班的航空公司，为 MU

actionCode:行动代码，OTHS 无需输入，赋值为空串。

person:OTHS 格式的输入规范，无需写入，因此此参数为 0

cityPair:OTHS 格式的输入规范无需输入，因此赋值为空串

airNo: OTHS 格式的输入规范无需输入，因此赋值为空串

fltClass: OTHS 格式的输入规范无需输入，因此赋值为空格字符

departureTime: OTHS 格式的输入规范无需输入，因此赋值为空串

serveInfo: OTHS 中这里输入需要注明的信息文本

psgrName：持卡人姓名，这条 SSR OTHS 无需注明该信息是某位旅客的特定信息，因此赋值为空串。体现在 SSR 中就是该条 SSR 不是以“/Pn”形式的字符串结尾。

③ 旅客提出希望在订取的航段（2009 年 6 月 29 日，MU5102，Y 舱，PEK 到 SHA）上申请靠窗座位的要求，为该旅客使用 SSR NSSW 进行申请，

SSR NSSW MU HN1 PEKSHA 5102 Y26JUN/P1

如上例 IBE 使用了以下代码进行添加

```
SellSeatexample.addSSR(new BookSSR("NSSW", airNo.substring(0,2),
        "nn", 1, orgCity+dstCity, airNo.substring(2), fltClass, departureTime, "", name));
```

serveCode:因为是靠窗座位申请，代码为 NSSW；

airCode:相应航段的执行航班的航空公司，为 MU

actionCode:行动代码，申请使用 NN。

person:NSSW 格式的输入规范，为一位旅客申请，使用 1

cityPair:相应航段的起飞到达机场代码对

airNo：相应航段的航班号，5102

fltClass：相应航段的订座舱位，Y

departureTime：相应航段的日期，格式为 yyyy-MM-dd,如上，2009-06-26

serveInfo：NSSW 的格式不允许输入其它的信息，因此使用空串

psgrName：这条 SSR NSSW 申请的旅客姓名，与订取 PNR 输入的旅客姓名一致，如例中，这条 SSR 是为名为 IBETESTER/IBEGROUP 申请的，因此这个参数使用这个旅客订座的名字。

④ 旅客提出希望在订取的航段（2009 年 6 月 29 日，MU5102，Y 舱，PEK 到 SHA）上申请无盐的特殊餐食的要求，在特殊餐食代码中只有低盐餐食，所有不能被特别的餐食代码覆盖的餐食申请都使用 SPML 申请，因此为该旅客使用 SSR SPML 进行申请，

SSR SPML MU HN1 PEKSHA 5102 Y26JUN NO SALT MEAL/P1

如上例 IBE 使用了以下代码进行申请

```
    SellSeatexample.addSSR(new BookSSR("SPML", airNo.substring(0,2),
        "nn", 1, orgCity+dstCity, airNo.substring(2), fltClass, departureTime, "NO SALT MEAL", name));
```

serveCode:因为是特殊餐食申请，代码为 SPML；

airCode:相应航段的执行航班的航空公司，为 MU

actionCode:行动代码，申请使用 NN。

person:NSSW 格式的输入规范，为一位旅客申请，使用 1

cityPair:相应航段的起飞到达机场代码对

airNo：相应航段的航班号，5102

fltClass：相应航段的订座舱位，Y

departureTime：相应航段的日期，格式为 yyyy-MM-dd,如上，2009-06-26

serveInfo：对特殊餐食的具体要求进行说明，这里注明无盐

psgrName：这条 SSR NSSW 申请的旅客姓名，与订取 PNR 输入的旅客姓名一致，如例中，这条 SSR 是为名为 IBETESTER/IBEGROUP 申请的，因此这个参数使用这个旅客订座的名字。

其它 SSR 的格式都类似以上例子给出，具体的 SSR 代码相当丰富，各自具有不同的意义，请参看订座系统手册。IBE 在 PNR 中添加特殊服务 SSR 的调用方式可以根据具体 SSR 代码的规定格式来设置相应的参数。

3.7.2 向 PNR 中添加 OSI

OSI 其它服务组，表示其他服务情况组提供服务的情况，是不需要立即回答的，相应的电报或 QUEUE 将会出现在有关的部门中.

当需要向 PNR 中添加 OSI 项时，IBE 的 SellSeat 提供了 public void addOSI(BookOSI osi)方法。

需要先自行构造一个 BookSSR 类，然后添加到订座信息中。

BookSSR 提供的构造函数如下:

public BookOSI(String airCode, String osi, String psgrName)

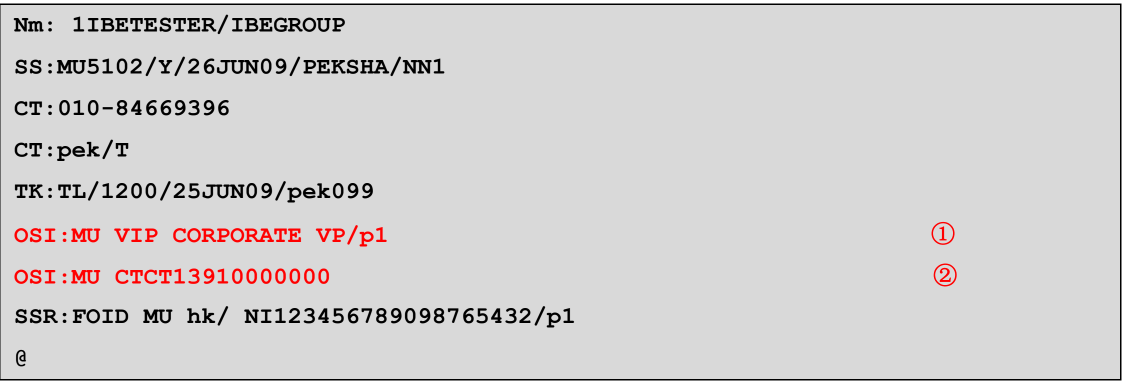
airCode:表示接收 OSI 的系统的代码，

osi:其它服务的信息

psgrName:如果专为某位旅客输入 OSI，则在这里指定旅客名字，与为旅客输入的姓名应一致。

旅客信息如 3.1 节，

主机指令为



```
Nm: 1IBETESTER/IBEGROUP
SS:MU5102/Y/26JUN09/PEKSHA/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/25JUN09/pek099
OSI:MU VIP CORPORATE VP/p1                        ①
OSI:MU CTCT13910000000                            ②
SSR:FOID MU hk/ NI123456789098765432/p1
@
```

输入后 PNR 如

```
1.IBETESTER/IBEGROUP XD49Y
 2.  MU5102 Y   FR26JUN  PEKSHA HK1   1300 1500         E T2--
 3.010-84669396
 4.PEK/T
 5.TL/1200/25JUN/PEK099
 6.SSR FOID MU HK1 NI123456789098765432/P1
 7.SSR ADTK 1E BY PEK23JUN09/1300 OR CXL MU5102 Y26JUN
 8.OSI MU VIP CORPORATE VP/P1                           ①
 9.OSI MU CTCT13910000000                               ②
10.RMK CA/ZY4Q4
11.PEK099
```

对应的 IBE 代码

```
    /*生成预定类对象*/
    SellSeat SellSeatexample= new SellSeat();

    /*添加旅客姓名*/
    String name="IBETESTER/IBEGROUP";
    SellSeatexample.addAdult(name);
```

```
/*添加旅客乘坐航段信息*/
String airNo="MU5102";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-06-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客身份证信息*/
String airline="MU";
String idtype="NI";
String id="123456789098765432";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);
/*添加 OSI 信息*/
String osiInfo = "VIP CORPORATE VP";
SellSeatexample.addOSI(new BookOSI(airline,osiInfo,name));
String osiInfo2 = "CTCT13910000000";
SellSeatexample.addOSI(new BookOSI(airline,osiInfo2,""));

/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-06-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

代码中的

① 使用 addOSI 方法添加了一个构造出来的 OSI。例中 OSI 表示为旅客 IBETESTER/IBEGROUP 建立 OSI 项，表明他的 VIP 信息。

② 使用 addOSI 方法添加一个联系信息，例中表示航空公司可以从该 OSI 中获取旅客联系信息以便在需要的时候联系旅客，通知航班动态等，因为不指定特定旅客，因此 name 参数使用空串。

注：为了便于航空公司联系旅客，航空公司要求以 CTC 开头的 OSI 信息输入真实有效的联系信息。

3.7.3 向 PNR 中添加 RMK

备注组可以记录某些可能有助于了解旅客的情况.

当需要添加备注组时，IBE 提供了方法　public void addRMK(BookRMK rmk) 使用此方法需要先自行构造一个 BookRMK 类，然后添加到订座信息中。

BookRMK 提供的构造函数如下:

public BookRMK(String rmktype, String rmkinfo, String psgrName)

rmktype:rmk 类型，大部分的 RMK 项没有类型，可以输入空串，如有特殊类型的 RMK，如手机号 MP，则可以在该参数设置。

rmkinfo:需要输入的 rmk 信息，无类型的 RMK 可以使用自由格式文本，有类型的 RMK 按相应类型的数据规范输入

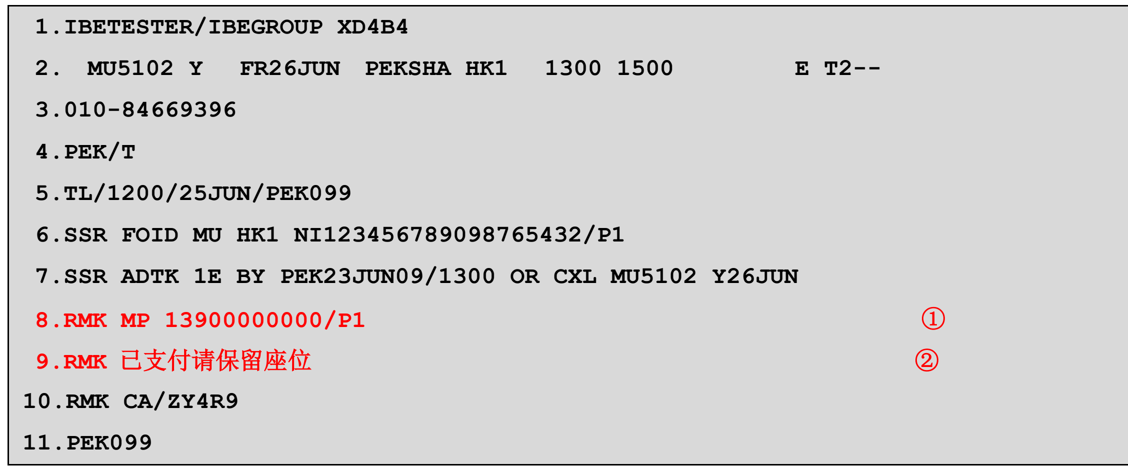
psgrName:如果该 RMK 是为 PNR 中特定旅客输入的，则设置为旅客姓名，否则，可设置为空串。

旅客信息如 3.1 节，

主机指令为

```
Nm: 1IBETESTER/IBEGROUP
SS:MU5102/Y/26JUN09/PEKSHA/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/25JUN09/pek099
SSR:FOID MU hk/ NI123456789098765432/p1
RMK:MP 13900000000/p1                                  ①
RMK:已支付请保留座位                                    ②
@
```

输入后 PNR 如



```
 1.IBETESTER/IBEGROUP XD4B4
 2.  MU5102 Y   FR26JUN  PEKSHA HK1   1300 1500        E T2--
 3.010-84669396
 4.PEK/T
 5.TL/1200/25JUN/PEK099
 6.SSR FOID MU HK1 NI123456789098765432/P1
 7.SSR ADTK 1E BY PEK23JUN09/1300 OR CXL MU5102 Y26JUN
 8.RMK MP 13900000000/P1                              ①
 9.RMK 已支付请保留座位                                ②
10.RMK CA/ZY4R9
11.PEK099
```

对应的 IBE 代码

```
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
String name="IBETESTER/IBEGROUP";
```

```
SellSeatexample.addAdult(name);

/*添加旅客乘坐航段信息*/
String airNo="MU5102";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-06-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客身份证信息*/
String airline="MU";
String idtype="NI";
String id="123456789098765432";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);

 /*添加 RMK 信息*/
String rmkType = "MP";
String rmkInfo = "13900000000";
SellSeatexample.addRMK(new BookRMK(rmkType,rmkInfo,name));
String rmkInfo2 = "已支付请保留座位";
SellSeatexample.addRMK(new BookRMK("" ,rmkInfo2,""));



/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-06-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

代码中的

① 使用 addRMK 方法添加了一个构造出来的 RMK。例中 RMK 表示为旅客 IBETESTER/IBEGROUP 建立 RMK MP 项，表明他的手机信息，并指明了他的姓名信息。航空公司或者代理人可以通过指令向该手机号发送行程短信

② 使用 addRMK 方法添加一个 RMK 信息，例中的 RMK 是输入的自由文本，RMK 项支持英文信息，中文信息，或中英文混编信息。该 RMK 因为没有必要指明特别的旅客信息，因此构造 RMK 时 psgrName 参数设置为空串。

3.7. 预定团队 PNR

3.8.1 生成一个申请座位的团队 PNR

➢ eTerm 预定 PNR 过程

eTerm 中运用主机指令预定旅客的订座记录如下：

```
GN:10TUANDUI                                                --①
SS:CA1501/Y/26MAY/PEKSHA/NN10
CT:010-84669396
TK:TL/1200/25MAY/PEK099
@
  CA1501  Y TU26MAY  PEKSHA DN10  0830 1040    6
T041Q  - ADD SSR TKNE FOR NEW ET SEGMENT 
```

与预定散客（9 人或以下）PNR 类似，需要输入航段，联系组，出票时限。团队 PNR 可以先输入团队人数和团队名，待以后再输入旅客信息。

① 输入团队名称与人数(NM)：可对应 setGroupTicket(boolean isGroup)方法和 setGroupName(String groupName) 方法、setPassengerNumber(int passengerNumber)方法。

首先 setGroupTicket(true)表示这是一个团队订座
然后 setGroupName(“TUANDUI”)设置团队名
然后 setPassengerNumber(10)设置团队人数

TIPS:

只有设置 GroupTicket 标识为真后，IBE 才会认为在订取一个团队，团名和团队人数的设置才会有效
设置团队名时： 团队命名规则请参考订座手册，并遵循航空公司的要求
对于代理人订座，团队人数不能少于 10 人，一旦确定团队人数，则不能再增加，如生成 PNR 后再需要增加人数，只能另行订取 PNR。但是可以通过分离团队或者删除旅客来缩减团队 PNR 人数，分离与删除旅客的限制必须遵从航空公司规定。
航段组可以用 NN 或者 LL 作为 actionCode 参数来申请团队座位。

产生的 PNR 结果：

```
0.10IBETUANDUI NM0 T041Q
1.  CA1501 Y   TU26MAY  PEKSHA HN10  0830 1040      E T3--
2.010-84669396
3.TL/1200/25MAY/PEK099
4.RMK CA/C0J90
5.PEK099
```

- IBE 实现团队 PNR 预定

```
/*
 * 团队预定示例
 *
 * */
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

String groupName = "TUANDUI";
int groupPassengerNumber = 10;

/* 设置团队信息*/
SellSeatexample.setGroupTicket(true);
SellSeatexample.setGroupName(groupName);
SellSeatexample.setPassengerNumber(groupPassengerNumber);

/*添加旅客乘坐航段信息*/
String airNo="CA1501";
char  fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=groupPassengerNumber;
String departureTime="2009-05-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);


/*添加团队联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加出票时限*/
String dateLimit="2009-05-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);
```

```
/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

3.8.2 在申请座位的同时输入旅客信息

团队订座生成 PNR 时，也可以同时输入部分或者全部旅客信息。

```
GN:10TUANDUI
NM: 1 旅客一 1 旅客二
SS:CA1501/Y/29MAY/PEKSHA/NN10
CT:010-84669396
TK:TL/1200/27MAY/PEK099
SSR:FOID CA HK/NI123456789/P1
SSR:FOID CA HK/NI223456789/P2
@
  CA1501  Y TU29MAY  PEKSHA DN10  0830 1040    6
T042G  - ADD SSR TKNE FOR NEW ET SEGMENT 
```

产生的 PNR 结果：

```
0.10TUANDUI NM2 T042G
1.旅客二 2.旅客一
3.  CA1501 Y   FR29MAY  PEKSHA HN10  0830 1040     E T3--
4.010-84669396
5.TL/1200/27MAY/PEK099
6.SSR FOID CA HK1 NI223456789/P1
7.SSR FOID CA HK1 NI123456789/P2
8.RMK CA/C0JKL
9.PEK099
```

如上，在生成 PNR 的时候即添加了两个旅客的信息,修改代码如下：

```
    /*
    * 团队预定示例
    *
    * */
    /*生成预定类对象*/
  SellSeat SellSeatexample= new SellSeat();


  String groupName = "TUANDUI";
```

```
int groupPassengerNumber = 10;

/*设置团队信息*/
SellSeatexample.setGroupTicket(true);
SellSeatexample.setGroupName(groupName);
SellSeatexample.setPassengerNumber(groupPassengerNumber);

 /*添加旅客姓名*/
 String name1="旅客一";
 SellSeatexample.addAdult(name1);
 String name2="旅客二";
 SellSeatexample.addAdult(name2);

 /*添加旅客身份证信息*/
 String airline="CA";
 String idtype="NI";
 String id="123456789";
 SellSeatexample.addSSR_FOID(airline,idtype,id,name1);
 String airline2="CA";
 String idtype2="NI";
 String id2="223456789";
 SellSeatexample.addSSR_FOID(airline2,idtype2,id2,name2);



/*添加旅客乘坐航段信息*/
String airNo="CA1501";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=groupPassengerNumber;
String departureTime="2009-05-29";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);


/*添加团队联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加出票时限*/
String dateLimit="2009-05-27 12:00:00";
SellSeatexample.setTimelimit(dateLimit);
```

```
/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult   ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

3.8. 订取国际成人单程 PNR

国际航段 PNR 订取上与国内并没有太大不同，但是根据各国的要求，旅客的身份信息由 API 定义，需要为每位旅客输入 SSR DOCS 来给出旅客身份信息，部分国家可能还需要为每位旅客输入 SSR DOCA，SSR DOCO。以美国航线为例，除了 DOCS，还需要输入 DOCA：

eTerm 中运用主机指令预定旅客的订座记录如下：

```
Nm: 1IBE/TEST
SS:CA983/Y/29DEC09/PEKLAX/NN1
CT:010-84669396
TK:TL/1200/27DEC09/pek099
SSR:DOCS CA hK1 P/CN/P0123456789/CN/04MAR70/F/30JUN10/IBE/TEST/H/p1
SSR:doca CA hk1 D/US/NO 11 TEST ROAD/LOS ANGELES/CA/10000/p1
@
D0418 -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
请及时出票, 自动出票时限: 09JUN09/1605
  CA 983  Y TU29DEC  PEKLAX DK1   2100 1730    6
  - ADD SSR TKNE FOR NEW ET SEGMENT
```

产生的 PNR 结果：

```
 1.IBE/TEST D0418
 2.  CA983  Y   TU29DEC  PEKLAX HK1   2100 1730         E T3--
 3.010-84669396
 4.PEK/T
 5.TL/1200/27DEC/PEK099
 6.SSR ADTK 1E BY PEK09JUN09/1616 OR CXL CA 983 Y29DEC
 7.SSR DOCS CA HK1 P/CN/P0123456789/CN/04MAR70/F/30JUN10/IBE/TEST/H/P1  ①
 8.SSR DOCA CA HK1 D/US/NO 11 TEST ROAD/LOS ANGELES/CA/10000/P1         ②
 9.RMK CA/T049G
10.PEK099
```

① 输入 SSR DOCS

对应 addSSR_DOCS(String airline, String doc_type, String issueCountry, String doc_No, String nationality,Date birth, String gender, boolean infant, Date expiry_Date, String surname, String firstname, String midname, boolean holder, String name)方法。

airline 承运人，与航段信息相关，如上例中，航班为 CA 代码航班，因此使用 CA
doc_type 类型，F/P/AC/A/C/I/IP，上例中为护照就使用 P
issueCountry 证件签发国，三字代码或两字代码，如中国护照可使用 CN 或 CHN，上例中为 CN
doc_No 证件号码，如上例中 P0123456789
nationality 旅客国籍，三字代码或两字代码均可，上例中为 CN
birth 出生日期，以证件上的记录为准，上例中为 04MAR70
gender 性别，一般 M 或者 F，上例中为 F
infant 婴儿标识，如果是为婴儿输入 DOCS，需设置为真，详见后面的成人带婴儿国际票例子。上例中设置为假，因此未显示。
expiry_Date 证件有效日期，上例中的 30JUN10，表示护照于 2010 年 6 月 30 日到期
surname 姓，需要与护照上的一致，(护照有扫描区的以扫描区为准)，上例中的 IBE
firstname 名字，需要与护照上的一致，(护照有扫描区的以扫描区为准)，上例中的 TEST
midname 需要与护照上的一致，(护照有扫描区的以扫描区为准)，可选（如果护照上的姓名没有中间名，则不填）
holder 护照持有者标志，如成人携带婴儿时，成人应当设置此标识，上例中的 H
name 姓名，与 PNR 中一致，婴儿使用携带人姓名

② 输入 SSR DOCA

对应 addSSR_DOCA(String airline, String doca_type, String country, String address, String city, String state, String zip, boolean infant, String name)方法。

SSR_DOCA 表示旅客的住址信息，美国要求，如前往美国的外籍人士，需要输入第一个目的地的地址。

airline 承运人，与航段信息相关，如上例中，航班为 CA 代码航班，因此使用 CA
doca_type 类型，“R”(居住住址)/“D”(目的地地址)
country 居住地/目的地的国家三字或者代码,USA/US/CHN...，上例中为 US
address 地址信息，35 字符之内，限 A-Z0-9a-z.,上例中为 NO 11 TEST ROAD。
city 城市，上例中为 LOS ANGELES
state 所在省/州二字代码，上例中为 CA
zip 邮政编码，上例中的 10000
infant 婴儿标识，如果是为婴儿输入 DOCA，需设置为真，详见后面的成人带婴儿国际票例子。上例中设置为假，因此未显示。
name 旅客姓名，婴儿使用携带人姓名

如上，在生成 PNR 的时候添加了旅客的 API(SSR DOCS 和 DOCA)信息,修改代码如下：

```
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();


/*添加旅客姓名*/
String name="IBE/TEST";
```

```
SellSeatexample.addAdult(name);

/*添加旅客身份信息--DOCS*/
/*添加一个 DOCS 信息在 IBE/TEST 名下*/
String airline="CA";
String doctype="P";
String issueCountry = "CN";
String doc_No = "P0123456789";
String nationality = "CN";
Date birth = new SimpleDateFormat("yyyyMMdd").parse("19700304");
String gender = "F";
boolean infant = false;
Date expireDate = new SimpleDateFormat("yyyyMMdd").parse("20100630");
String surname = "IBE";
String firstname = "TEST";
String midname = "";
boolean holder = true;

SellSeatexample.addSSR_DOCS(airline,doctype,issueCountry,doc_No,nationality,
        birth,gender,infant,expireDate,surname,firstname,midname,holder,name);



/*添加旅客身份信息--DOCA*/
/*添加一个 DOCA 信息在 IBE/TEST 名下,这里的 airline 和 infant 沿用了上面定义的*/
String doca_type = "D";
String country = "US";
String address = "NO 11 TEST ROAD";
String city = "LOS ANGELES";
String state = "CA";
String zip = "10000";
SellSeatexample.addSSR_DOCA(airline,doca_type,country,address,city,state,zip,infant,name);


/*添加旅客乘坐航段信息*/
String airNo="CA983";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="LAX";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-12-29";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);
```

```
/*添加联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加出票时限*/
String dateLimit="2009-12-27 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

TIPS：

不是所有国家都要求输入 DOCA，但是 SSR DOCS 一般都需要输入。具体应看当时的各国政府规定进行调整。

3.9. 订取国际成人加儿童单程 PNR

与成人国际 PNR 类似，儿童也要为其添加 API 信息，所需要的种类和成人是一致的，以美国的要求为例：eTerm 中运用主机指令预定旅客的订座记录如下：

```
Nm: 1IBE/TEST 1IBE/CHILD CHD
SS:CA983/Y/29DEC09/PEKLAX/NN2
CT:010-84669396
CT:pek/T
TK:TL/1200/27DEC09/pek099
SSR:DOCS CA hK1 P/CN/P0123456789/CN/04MAR70/F/30JUN10/IBE/TEST/H/p1
SSR:DOCS CA hK1 P/CN/P0123456790/CN/06APR03/M/30JUN10/IBE/CHILD/H/p2
SSR:doca CA hk1 D/US/NO 11 TEST ROAD/LOS ANGELES/CA/10000/p1
SSR:doca CA hk1 D/US/NO 11 TEST ROAD/LOS ANGELES/CA/10000/p2
@
D048M -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA 983  Y TU29DEC  PEKLAX DK2   2100 1730
 航空公司使用自动出票时限，请检查 PNR
```

产生的 PNR 结果：

```
 1.IBE/CHILD CHD 2.IBE/TEST D048M
 3.  CA983  Y   TU29DEC  PEKLAX HK2   2100 1730         E T3--
 4.010-84669396
 5.PEK/T
 6.TL/1200/27DEC/PEK099
 7.SSR ADTK 1E BY PEK08JUN09/1552 OR CXL CA 983 Y29DEC
 8.SSR DOCS CA HK1 P/CN/P0123456790/CN/06APR03/M/30JUN10/IBE/CHILD/H/P1
 9.SSR DOCS CA HK1 P/CN/P0123456789/CN/04MAR70/F/30JUN10/IBE/TEST/H/P2
10.SSR DOCA CA HK1 D/US/NO 11 TEST ROAD/LOS ANGELES/CA/10000/P1
```

```
11.SSR DOCA CA HK1 D/US/NO 11 TEST ROAD/LOS ANGELES/CA/10000/P2
12.RMK CA/T04NL
13.PEK099
```

与成人相比，儿童需要输入的信息是基本一致的，只需要按儿童的方式添加旅客即可。在生成 PNR 的时候添加了儿童旅客的 API(SSR DOCS 和 DOCA)信息,修改代码如下:

```
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();


/*添加旅客姓名*/
String name="IBE/TEST";
SellSeatexample.addAdult(name);
String child_name = "IBE/CHILD";
SellSeatexample.addChild(child_name);

/*添加旅客身份信息--DOCS*/
/*添加一个 DOCS 信息在 IBE/TEST 名下*/
String airline="CA";
String doctype="P";
String issueCountry = "CN";
String doc_No = "P0123456789";
String nationality = "CN";
Date birth = new SimpleDateFormat("yyyyMMdd").parse("19700304");
String gender = "F";
boolean infant = false;
Date expireDate = new SimpleDateFormat("yyyyMMdd").parse("20100630");
String surname = "IBE";
String firstname = "TEST";
String midname = "";
boolean holder = true;

SellSeatexample.addSSR_DOCS(airline,doctype,issueCountry,doc_No,nationality,
        birth,gender,infant,expireDate,surname,firstname,midname,holder,name);

/*添加一个 DOCS 信息在 IBE/CHILD 名下*/
String airline_c="CA";
String doctype_c="P";
String issueCountry_c = "CN";
String doc_No_c = "P0123456790";
String nationality_c = "CN";
```

```
        Date birth_c = new SimpleDateFormat("yyyyMMdd").parse("20030406");
        String gender_c = "M";
        boolean infant_c = false;
        Date expireDate_c = new SimpleDateFormat("yyyyMMdd").parse("20100630");
        String surname_c = "IBE";
        String firstname_c = "CHILD";
        String midname_c = "";
        boolean holder_c = true;

        SellSeatexample.addSSR_DOCS(airline_c,doctype_c,issueCountry_c,doc_No_c,nationality_c,
                birth_c,gender_c,infant_c,expireDate_c,surname_c,firstname_c,
                midname_c,holder_c,child_name);

       /*添加旅客身份信息--DOCA*/
       /*添加一个 DOCA 信息在 IBE/TEST 名下，因为同一个人的 airline 和 infant 标识是同样的，这里的
airline 和 infant 沿用了上面定义的*/
        String doca_type = "D";
        String country = "US";
        String address = "NO 11 TEST ROAD";
        String city = "LOS ANGELES";
        String state = "CA";
        String zip = "10000";
        SellSeatexample.addSSR_DOCA(airline,doca_type,country,address,city,state,zip,infant,name);

        /*添加一个 DOCA 信息在 IBE/CHILD 名下，因为同一个人的 airline 和 infant 标识是同样的，这
里的 airline 和 infant 沿用了上面定义的*/
        String doca_type_c = "D";
        String country_c = "US";
        String address_c = "NO 11 TEST ROAD";
        String city_c = "LOS ANGELES";
        String state_c = "CA";
        String zip_c = "10000";
        SellSeatexample.addSSR_DOCA(airline_c,doca_type_c,country_c,address_c,city_c,
              state_c,zip_c,infant_c,child_name);


       /*添加旅客乘坐航段信息*/
       String airNo="CA983";
       char    fltClass='Y';
       String orgCity="PEK";
       String dstCity="LAX";
       String actionCode="NN";
       int tktNum=2;
       String departureTime="2009-12-29";
```

```
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);


/*添加团队联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加出票时限*/
String dateLimit="2009-12-27 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

3.10. 订取国际成人加婴儿单程 PNR

成人的订取逻辑和原来一样，通过 addInfant 方法添加了婴儿组。对于婴儿，也需要为他添加 API 信息，但是不同的是，对应的 API 信息（DOCS,DOCA）中的 infant 标识设为真，而相应的 holder 标识不设置，同时，API 信息的姓名仍然应该是携带此婴儿的成人旅客的姓名。

在 eTerm 里的主机操作如下：

```
Nm: 1IBE/TEST
SS:CA983/Y/29DEC09/PEKLAX/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/27DEC09/pek099
XN:IN/IBE/INFANT INF(JAN09)/p1                                    ①
SSR:DOCS CA hK1 P/CN/P0123456789/CN/04MAR70/F/30JUN10/IBE/TEST/H/p1
SSR:DOCS CA hK1 P/CN/P0123456790/CN/03JAN09/MI/30JUN10/IBE/INFANT/p1 ②
SSR:doca CA hk1 D/US/NO 11 TEST ROAD/LOS ANGELES/CA/10000/p1
SSR:doca CA hk1 D/US/NO 11 TEST ROAD/LOS ANGELES/CA/10000/I/p1     ③
SSR:INFt CA nn1 PEKLAX 983 Y 29DEC09 IBE/INFANT 03JAN09/p1         ④
@
D0499 -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA 983  Y TU29DEC  PEKLAX DK1   2100 1730
 航空公司使用自动出票时限，请检查 PNR
```

⑤ 输入婴儿信息，对应 addInfant(String birth, String carrierName, String name)方法。

⑥ 输入婴儿的 API 信息 DOCS，与成人添加 DOCS 对应的方法是一样的，但是要注意：infant 参数必须设置为真，而最后的 name 参数应该是携带此旅客一同旅行的成人的名字。此方法中的 surname，firstname，midname 等参数应当与输入婴儿信息时的婴儿姓名中对应部分一致。

⑦ 输入婴儿的 API 信息 DOCA，与成人添加 DOCA 的方法是一致的，但是 infant 参数设置为真，而最后的 name 参数应该是携带此旅客一同旅行的成人的名字。

⑧ 输入婴儿的 SSR INFT 申请，某些航空公司要求订座时为**每个**婴儿的**每个**已经确定的航段输入 SSR INFT 项进行申请。输入此项可对应方法 addSSR_INFT(String airline, String segment, String fltNo, String fltClass, String date, String infantName, String infantBirth, String freeText,String carrierName) 方法。如果航空公司要求输入此项必须该申请获得航空公司批准，即该 SSR 由申请时的 NN 状态变成 KK 或者 HK 状态，之后才可以为婴儿出票。批准的过程可能是自动的也可能是手动的。因此，可能生成 PNR 后即已状态变更为 KK，也可能需要一段时间或者营业员主动联系后才会有状态变更，如果状态一直为 NN 或者被变更为 NO，请不要为婴儿出票，同行成人是否还需要单独出票，需要再次联系旅客。

airline:该航段承运人的两字代码，如例中的“CA”，

segment:该航段的起始到达地三字代码对，如“PEKLAX”

fltNo:该航班的航班号，不带航空公司代码，如“983”

fltClass:同行成人旅客的订座舱位，如“Y”

date:旅行日期,即相应航段日期,如“29DEC”,格式为“DDMMM”/“DDMMMYY”

infantName:婴儿姓名,格式为 XX/XXX,不接受中文姓名.如此例中“IBE/INFANT”

infantBirth:婴儿生日，格式“DDMMYY”,如“03JAN09”

freeText:如有额外需要的注意的，可写在这里，如果没有，设置为空串

carrierName:同行携带此旅客的成人姓名，应当与订座时输入的旅客姓名完全一致

产生的 PNR 结果：

```
 1.IBE/TEST D0499
 2.  CA983  Y  TU29DEC  PEKLAX HK1   2100 1730         E T3--
 3.010-84669396
 4.PEK/T
 5.TL/1200/27DEC/PEK099
 6.SSR ADTK 1E BY PEK08JUN09/1713 OR CXL CA 983 Y29DEC
 7.SSR DOCS CA HK1 P/CN/P0123456790/CN/03JAN09/MI/30JUN10/IBE/INFANT/P1
 8.SSR DOCS CA HK1 P/CN/P0123456789/CN/04MAR70/F/30JUN10/IBE/TEST/H/P1
 9.SSR DOCA CA HK1 D/US/NO 11 TEST ROAD/LOS ANGELES/CA/10000/I/P1
10.SSR DOCA CA HK1 D/US/NO 11 TEST ROAD/LOS ANGELES/CA/10000/P1
11.SSR INFT CA KK1 PEKLAX 983 Y29DEC IBE/INFANT 03JAN09/P1
12.OSI YY 1INF IBE/INFANT INF/P1
13.RMK CA/T04PT
14.XN/IN/IBE/INFANT INF(JAN09)/P1
15.PEK099
```

在录入婴儿的同时，为婴儿输入了 API 信息：SSR DOCA 和 SSR DOCS，同时为婴儿输入了 SSR INFT 请求：

```
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();
```

```
    /*添加旅客姓名*/
    String name="IBE/TEST";
    SellSeatexample.addAdult(name);
    /*添加婴儿的基本信息*/
    String infant_name = "IBE/INFANT";
    String infant_birth = "2009-01-03";
    SellSeatexample.addInfant(infant_birth,name,infant_name);

   /*添加旅客乘坐航段信息*/
   String airNo="CA983";
   char    fltClass='Y';
   String orgCity="PEK";
   String dstCity="LAX";
   String actionCode="NN";
   int tktNum=1;
   String departureTime="2009-12-29";
   SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

    /*添加旅客身份信息--DOCS*/
    /*添加一个 DOCS 信息在 IBE/TEST 名下*/
    String airline="CA";
    String doctype="P";
    String issueCountry = "CN";
    String doc_No = "P0123456789";
    String nationality = "CN";
    Date birth = new SimpleDateFormat("yyyyMMdd").parse("19700304");
    String gender = "F";
    boolean infant = false;
    Date expireDate = new SimpleDateFormat("yyyyMMdd").parse("20100630");
    String surname = "IBE";
    String firstname = "TEST";
    String midname = "";
    boolean holder = true;

    SellSeatexample.addSSR_DOCS(airline,doctype,issueCountry,doc_No,nationality,
          birth,gender,infant,expireDate,surname,firstname,midname,holder,name);

    /*添加一个 DOCS 信息在 IBE/INFANT 名下*/
    String airline_i="CA";
    String doctype_i="P";
    String issueCountry_i = "CN";
    String doc_No_i = "P0123456790";
    String nationality_i = "CN";
```

```
            Date birth_i = new SimpleDateFormat("yyyyMMdd").parse("20090103");
            String gender_i = "M";
            boolean infant_i = true;
            Date expireDate_i = new SimpleDateFormat("yyyyMMdd").parse("20100630");
            String surname_i = "IBE";
            String firstname_i = "INFANT";
            String midname_i = "";
            boolean holder_i = false;

            SellSeatexample.addSSR_DOCS(airline_i,doctype_i,issueCountry_i,doc_No_i,nationality_i,
                  birth_i,gender_i,infant_i,expireDate_i,surname_i,firstname_i,midname_i,holder_i,name);

           /*添加旅客身份信息--DOCA*/
           /*添加一个 DOCA 信息在 IBE/TEST 名下，因为同一个人的 airline 和 infant 标识是同样的，这里的
airline 和 infant 沿用了上面定义的*/
            String doca_type = "D";
            String country = "US";
            String address = "NO 11 TEST ROAD";
            String city = "LOS ANGELES";
            String state = "CA";
            String zip = "10000";
            SellSeatexample.addSSR_DOCA(airline,doca_type,country,address,city,state,zip,infant,name);

            /*添加一个 DOCA 信息在 IBE/INFANT 名下，因为同一个人的 airline 和 infant 标识是同样的，这
里的 airline 和 infant 沿用了上面定义的*/
            String doca_type_i = "D";
            String country_i = "US";
            String address_i = "NO 11 TEST ROAD";
            String city_i = "LOS ANGELES";
            String state_i = "CA";
            String zip_i = "10000";
            SellSeatexample.addSSR_DOCA(airline_i,doca_type_i,country_i,
                        address_i,city_i,state_i,zip_i,infant_i,name);



           /*为婴儿的第 0 航段添加 SSR INFT */
           String airline0_i = "CA";
           String segment0_i = "PEKLAX";
           String fltNo0_i = "983";
           String fltclass0_i = "Y";
           String date0_i = "29DEC09";
           String infantName = "IBE/INFANT";
           String infantBirth = "03JAN09";
```

```
String freeText = "";
SellSeatexample.addSSR_INFT(airline0_i, segment0_i, fltNo0_i, fltclass0_i,
        date0_i, infantName, infantBirth, freeText, name);


/*添加团队联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加出票时限*/
String dateLimit="2009-12-27 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

3.11. 订取成人单程含运价 PNR

在出票前，为自动结算，需要在 PNR 输入一些数据，如 FN/FC/FP，PNR 中 FN 表示 PNR 中对应该 FN 的旅客的价格信息，包括票面价格，实付价格，代理费，税等。FN 里的价格均为单人价格，而不是 PNR 中所有或者多个旅客的价格之和。PNR 中的 FC 表示对应于该 FC 的旅客的运价计算，里面列出了 FN 中的运价是如何计算得到的，以及 XT 税拆分，以及在 EI 中写不下的运价使用条件等。FC 的运价计算中包括了航段信息，因此 FC 的输入要求其行程与 PNR 当前的航段组信息是一致的。FP 表示对应于该 FC 的旅客的支付方式。
在标准舱位 F/C/Y 之外的舱位一般要求输入运价使用的条件 EI，某些运价还要求输入旅游代码 TC。现在 BSP 国内票航空公司一般都要求使用 PAT:A 录入，但是国际票或者航空公司可能还是需要加入一些特别的信息，还是需要手工录入。IBE 提供了手工录入的接口来输入 FN/FC/FP/TC/EI。

```
录入 FN: public void addFN(BookFN fn)
         public void addFN(String fnStr)
录入 FC: public void addFC(BookFC fc)
         public void addFC(String fcStr)
录入 FP: public void addFP(BookFP fp)
         public void addFP(String fpStr)
录入 TC: public void addTC(BookTC tc)
录入 EI: public void addEI(BookEI ei)
```

根据 3.1 的例子里的旅客信息一次性生成含 FN/FC/FP/TC/EI 的 PNR：
主机操作如下

```
Nm: 1IBETESTER/IBEGROUP
SS:MU5102/Y/26JUN09/PEKSHA/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/25JUN09/pek099
SSR:FOID MU hk/ NI123456789098765432/p1
EI:不得签转                                              ⑤
TC:F/ARWNN                                              ④
FN:FCNY1130.00/SCNY1130.00/C0.00/TCNY50.00CN/TEXEMPTYQ  ①
FC:PEK MU SHA 1130.00Y CNY1130.00END                    ②
FP:CASH,CNY                                             ③
@
  MU5102  Y FR26JUN  PEKSHA NN1   0800 1000
XD404
 □航空公司使用自动出票时限，请检查 PNR□
```

输入了 FN/FC/FP/TC/EI。

产生的 PNR 结果：

```
1.IBETESTER/IBEGROUP XD404
 2.  MU5102 Y   FR26JUN  PEKSHA HN1   0800 1000         E T2--
 3.010-84669396
 4.PEK/T
 5.TL/1200/25JUN/PEK099
 6.FC/PEK MU SHA 1130.00Y CNY1130.00END
 7.SSR FOID MU HK1 NI123456789098765432/P1
 8.RMK CA/T045T
 9.FN/FCNY1130.00/SCNY1130.00/C0.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY1180.
 00
10.TC/ARWNN
11.EI/不得签转
12.FP/CASH,CNY
13.PEK099
```

① 用文本的方式输入 FN，先构造一个 BookFN 对象，然后使用 BookFN 的 setFN(String newFN) 等方法设置 FN 文本及其它参数，再使用 SellSeat 的 addFN(BookFN fn) 方法添加到 pnr

② 用文本的方式输入 FC，先构造一个 BookFC 对象，然后使用 BookFC 的 setFC(String newFC) 等方法设置 FC 文本及其它参数，再使用 SellSeat 的 addFC(BookFC fc) 方法添加到 pnr

③ 用文本的方式输入 FP，先构造一个 BookFP 对象，然后使用 BookFP 的 setFP(String newFP) 等方法设置 FP 文本及其它参数，再使用 SellSeat 的 addFP(BookFP fp) 方法添加到 pnr

④ 输入 TC，先使用构造自由文本格式的 TC 的方法 BookTC（String tcCode）构造。然后使用 SellSeat 的 public void addTC(BookTC tc)添加。

TC 还有一种标准格式的输入方法，国内客票使用不多，这种 TC 有严格的格式。如果要设置这种格式的 TC，在任意构造 BookTC 后使用 setTC 方法设置。在设置的标准格式 TC 不为空（null 或空串）的情况下，将不会使用自由文本格式的 TC，无论其取值如何。

TC 可以指定为某位乘客专用

任何情况下，添加的 TC 不能自由文本和标准格式 TC 都是空串或者 null。

⑤ 输入 EI，先使用 EI 的构造函数 BookEI(String endorsement)构造后使用 SellSeat 的 public void addEI(BookEI ei)方法添加。

EI 的内容就是一段文本，长度受主机限制。EI 可以指定为某位乘客专用。

任何情况下，EI 的文本内容不能为空串或者 null。

IBE 生成这样的 PNR 的代码如后

```
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加旅客乘坐航段信息*/
String airNo="MU5102";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-06-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客身份证信息*/
String airline="MU";
String idtype="NI";
String id="123456789098765432";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);


/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-06-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*FN*/
```

```
BookFN fn = new BookFN();
fn.setFn("FCNY1130.00/SCNY1130.00/C0.00/TCNY50.00CN/TEXEMPTYQ");
SellSeatexample.addFN(fn);

/*FC*/
BookFC fc = new BookFC();
fc.setFc("PEK MU SHA 1130.00Y CNY1130.00END");
SellSeatexample.addFC(fc);

/*FP*/
BookFP fp = new BookFP("CASH,CNY","");
SellSeatexample.addFP(fp);

/*EI*/
BookEI ei = new BookEI("不得签转","");
SellSeatexample.addEI(ei);

/*TC*/
BookTC tc = new BookTC();
tc.setFreecode("ARWNN");
SellSeatexample.addTC(tc);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult   ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

FN，FC 也可以使用格式化数据的方式输入。同上信息，FN，FC 改为

```
/*FN*/
BookFN fn = new BookFN();
//fn.setFn("FCNY1130.00/SCNY1130.00/C0.00/TCNY50.00CN/TEXEMPTYQ");
fn.setAmount(BookFN.F,"CNY",1130);                              ①
fn.setAmount(BookFN.S,"CNY",1130);                              ②
fn.setC(0);                                                     ③
fn.addTax(BookFN.T,"CNY",50,"CN");                              ④
fn.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"YQ");                   ⑤
SellSeatexample.addFN(fn);

/*FC*/
BookFC fc = new BookFC();
//fc.setFc("PEK MU SHA 1130.00Y CNY1130.00END");
fc.addFC("PEK","SHA","MU","Y",1130,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
```

```
//fc.addFC("PEK","SHA","MU","Y",1130,null,null,null,false,true,null,-1,null,-1,null,false,false,null,null);
            fc.setMoneytype("CNY");
            SellSeatexample.addFC(fc);
```

生成的指令和本例中用文本方式输入 FN/FC 得到的 pnr 是一样的。fc.addFC 行和注释掉的 fc.addFC 行也是一致的，fc 的输入可选的参数较多，具体参数的意义请参照 BSP 出票手册、国际订座手册和 BookFC 的相应文档。

① setAmount 方法有三个参数，

第一个表明需要设置 FN 中的哪个 FIELD，常见的包括 F 表示票面，S 表示实收，E 表示等值货币。税项 T 不使用这个方法，而是使用 addTax 方法。代理费率不使用这个方法，参见③。总合 A 项，税总和 X 项，在换开等情形下需要输入的 R 项，以及比较少见的 B 项也是可以接受的参数。一般情况下，总和 A 和税费总合 X 不需要输入，由系统自动计算就可以了。

第二个参数为这个 FIELD 的货币类型，

第三个参数表示货币数值。例中这一行设置票面价为 CNY1130。

② 这一行设置实收价为 CNY1130。

③ 这一行设置代理费率为 0。该值在 0-100 之间

④ 这一行添加了一个值为 CNY50 的代码为 CN 的税。addTax 方法的四个参数：

第一个为税项类型，常见的 T 表示输入此次开票收的税款，其他可接受的参数有 R 表示换开时退还税，O 表示换开时输入的原票税款；

第二个参数为货币代码，

第三个参数为货币数值，

第四个参数为税代码。

⑤ 这一行添加了一个值为豁免的代码为 YQ 的税。如果税值为 0，但是又必须输入，则使用常量 EXEMPTTAX 输入。系统不接受值为 0 的税。如果税值为豁免，则该税货币代码可以设置为空串。

3.12. 预订成人往返程含运价 PNR

类似于单程，但是 FN 里的价格应该是两段的总价格，FC 按照航段给出运价计算横式。以下分两种情况给出例子：

3.13.1 运价为两单程简单加和

国内客票标准舱位一般往返程运价为两段的加和，主机操作如下

```
Nm: 1IBETESTER/IBEGROUP
SS:MU5102/Y/26JUN09/PEKSHA/NN1
SS:MU5101/Y/29DEC09/SHAPEK/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/25JUN09/pek099
SSR:FOID MU hk/ NI123456789098765432/p1
EI:不得签转
TC:F/ARWNN
FN:FCNY2260.00/SCNY2260.00/c0.00
-   /tCNY100.00CN/tExemPtYQ
FC:PEK    MU SHA 1130.00Y
-    MU PEK 1130.00Y
- CNY2260.00END
FP:CASH,CNY
@
  MU5102  Y FR26JUN  PEKSHA DK1   0800 1000
  MU5101  Y TU29DEC  SHAPEK DK1   0800 0955
XD41R -  □航空公司使用自动出票时限, 请检查 PNR□
```

输入了 FN/FC/FP/TC/EI。

产生的 PNR 结果:

```
1.IBETESTER/IBEGROUP XD41R
 2.  MU5102 Y   FR26JUN  PEKSHA HK1   0800 1000         E T2--
 3.  MU5101 Y   TU29DEC  SHAPEK HK1   0800 0955         E --T2
 4.010-84669396
 5.PEK/T
 6.TL/1200/25JUN/PEK099
 7.FC/PEK MU SHA 1130.00Y MU PEK 1130.00Y CNY2260.00END
 8.SSR FOID MU HK1 NI123456789098765432/P1
 9.SSR ADTK 1E BY PEK23JUN09/0800 OR CXL MU5102 Y26JUN
10.RMK CA/ZY44K
11.FN/FCNY2260.00/SCNY2260.00/C0.00/XCNY100.00/TCNY100.00CN/TEXEMPTYQ/
    ACNY2360.00
12.TC/ARWNN
13.EI/BUDEQIANZHUAN 不得签转
14.FP/CASH,CNY
15.PEK099
```

IBE 生成这样的 PNR 的代码如后

```
        /*生成预定类对象*/
        SellSeat SellSeatexample= new SellSeat();

        /*添加旅客姓名*/
```

```
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加旅客乘坐航段信息*/
String airNo="MU5102";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-06-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客乘坐返程航段信息*/
String airNo2="MU5101";
char    fltClass2='Y';
String orgCity2="SHA";
String dstCity2="PEK";
String actionCode2="NN";
int tktNum2=1;
String departureTime2="2009-12-29";
SellSeatexample.addAirSeg(airNo2,fltClass2,orgCity2,dstCity2,actionCode2,tktNum2,departureTime2);


/*添加旅客身份证信息*/
String airline="MU";
String idtype="NI";
String id="123456789098765432";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);


/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-06-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*FN*/
BookFN fn = new BookFN();
//fn.setFn("FCNY2260.00/SCNY2260.00/C0.00/TCNY100.00CN/TEXEMPTYQ");
fn.setAmount(BookFN.F,"CNY",2260);
fn.setAmount(BookFN.S,"CNY",2260);
```

```
            fn.setC(0);
            fn.addTax(BookFN.T,"CNY",100,"CN");
            fn.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"YQ");
            SellSeatexample.addFN(fn);

            /*FC*/
            BookFC fc = new BookFC();
            //fc.setFc("PEK MU SHA 1130.00Y MU PEK 1130.00Y CNY2260.00END");          ①
            fc.addFC("PEK","SHA","MU","Y",1130,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);   ②
            fc.addFC("SHA","PEK","MU","Y",1130,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
                                                                                   ②’
          //fc.addFC("PEK","SHA","MU","Y",1130,null,null,null,false,true,null,-1,null,-1,null,false,false,null,null);
          //fc.addFC("SHA","PEK","MU","Y",1130,null,null,null,false,true,null,-1,null,-1,null,false,false,null,null);
            fc.setMoneytype("CNY");
            SellSeatexample.addFC(fc);

            /*FP*/
            BookFP fp = new BookFP("CASH,CNY","");
            SellSeatexample.addFP(fp);

            /*EI*/
            BookEI ei = new BookEI("不得签转","");
            SellSeatexample.addEI(ei);

            /*TC*/
            BookTC tc = new BookTC();
            tc.setFreecode("ARWNN");
            SellSeatexample.addTC(tc);

            /*完成 PNR 必需信息输入，递交主机，生成 PNR*/
            SSResult    ssr=SellSeatexample.commit1();

            /*PNR 结果*/
            System.out.println(ssr.getPnrno());
```

如上①②两种方式，①文本输入，②结构化输入，是等效的。

3.13.2 运价为特定的往返程运价

一种比较常见的情况就是航空公司对来回程一并订取的 PNR，或者中转联程的 PNR 有特别的优惠运价，较之两航段的公布运价简单加和有优惠，但是必须在出票时体现。假设有一个航段如上的运价，以 YRT 订取可以获得优惠运价为 2000，运价手工录入的例子和 IBE 代码如下：

```
Nm: 1IBETESTER/IBEGROUP
SS:MU5102/Y/26JUN09/PEKSHA/NN1
SS:MU5101/Y/29DEC09/SHAPEK/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/25JUN09/pek099
SSR:FOID MU hk/ NI12345678909876543 2/p1
EI:不得签转
TC:F/ARWNN
FN:FCNY2000.00/SCNY2000.00/c0.00
-   /tCNY100.00CN/tExemPtYQ
FC:PEK    MU SHA
-    MU PEK 2000.00YRT
- CNY2000.00END
FP:CASH,CNY
@
  MU5102  Y FR26JUN  PEKSHA DK1   0800 1000
  MU5101  Y TU29DEC  SHAPEK DK1   0800 0955
XD41R -  □航空公司使用自动出票时限，请检查 PNR□
```

输入了 FN/FC/FP/TC/EI。

产生的 PNR 结果：

```
1.IBETESTER/IBEGROUP XD41Z
 2.  MU5102 Y  FR26JUN  PEKSHA HK1   0800 1000        E T2--
 3.  MU5101 Y  TU29DEC  SHAPEK HK1   0800 0955        E --T2
 4.010-84669396
 5.PEK/T
 6.TL/1200/25JUN/PEK099
 7.FC/PEK MU SHA MU PEK 2000.00YRT CNY2000.00END
 8.SSR FOID MU HK1 NI12345678909876543 2/P1
 9.SSR ADTK 1E BY PEK23JUN09/0800 OR CXL MU5102 Y26JUN
10.RMK CA/ZY46Z
11.FN/FCNY2000.00/SCNY2000.00/C0.00/XCNY100.00/TCNY100.00CN/TEXEMPTYQ/
    ACNY2100.00
12.TC/ARWNN
13.EI/BUDEQIANZHUAN 不得签转
14.FP/CASH,CNY
15.PEK099
```

IBE 生成这样的 PNR 的代码如后

```
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
```

```
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加旅客乘坐航段信息*/
String airNo="MU5102";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-06-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客乘坐返程航段信息*/
String airNo2="MU5101";
char    fltClass2='Y';
String orgCity2="SHA";
String dstCity2="PEK";
String actionCode2="NN";
int tktNum2=1;
String departureTime2="2009-12-29";
SellSeatexample.addAirSeg(airNo2,fltClass2,orgCity2,dstCity2,actionCode2,tktNum2,departureTime2);


/*添加旅客身份证信息*/
String airline="MU";
String idtype="NI";
String id="123456789098765432";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);


/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-06-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*FN*/
BookFN fn = new BookFN();
//fn.setFn("FCNY2000.00/SCNY2000.00/C0.00/TCNY100.00CN/TEXEMPTYQ");
fn.setAmount(BookFN.F,"CNY",2000);
fn.setAmount(BookFN.S,"CNY",2000);
```

```
fn.setC(0);
fn.addTax(BookFN.T,"CNY",100,"CN");
fn.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"YQ");
SellSeatexample.addFN(fn);

/*FC*/
BookFC fc = new BookFC();
//fc.setFc("PEK MU SHA 1130.00Y MU PEK 1130.00Y CNY2260.00END");              ①

fc.addFC("PEK","SHA","MU",null,-1,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);  ②
fc.addFC("SHA","PEK","MU","YRT",2000,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);

//fc.addFC("PEK","SHA","MU","Y",-1,null,null,null,false,true,null,-1,null,-1,null,false,false,null,null);
//fc.addFC("SHA","PEK","MU","YRT",2000,null,null,null,false,true,null,-1,null,-1,null,false,false,null,null);
fc.setMoneytype("CNY");
SellSeatexample.addFC(fc);

/*FP*/
BookFP fp = new BookFP("CASH,CNY","");
SellSeatexample.addFP(fp);

/*EI*/
BookEI ei = new BookEI("不得签转","");
SellSeatexample.addEI(ei);

/*TC*/
BookTC tc = new BookTC();
tc.setFreecode("ARWNN");
SellSeatexample.addTC(tc);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

如上，当 fc.addFC()方法录入 FC 中的一个航段时，如果运价设置为-1，表示此航段与之后的一个或多个航段形成捆绑的运价，所有捆绑的航段运价将在这些航段的最后一个给出。如例子中，两个航段捆绑为 YRT 运价，在两航段的最后航段给出 PEK-SHA-PEK 的运价为 2000，运价基础为 YRT。

这种输入方式也使用于航空公司的两段或者多段联程运价，如假设 SHE-WUH-CTU 有 MU 的联程运价为 1000，运价基础为 YCT 则 FC 举例为：

SHE MU WUH MU CTU 1000.00YCT CNY1000.00END

IBE 生成这样的 PNR 的代码如后

```
fc.addFC("SHE","WUH","MU",null,-1,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc.addFC("WUH","CTU","MU","YCT",1000,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
```

如果是这样的联程提供来回程，FC 格式举例为：
SHE MU WUH MU CTU 1000.00YCT MU WUH MU SHE 1000.00YCT CNY2000.00END
IBE 生成这样的 PNR 的代码如后

```
fc.addFC("SHE","WUH","MU",null,-1,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc.addFC("WUH","CTU","MU","YCT",1000,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);

fc.addFC("CTU","WUH","MU",null,-1,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc.addFC("WUH","CTU","MU","YCT"， 1000,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
```

如果多段联程，如 SHE-WUH-CTU-KMG，仍然是 YCT，价格为 1200，则 FC 为：
SHE MU WUH MU CTU MU KMG 1200.00YCT CNY1000.00END
IBE 代码如例：

```
fc.addFC("SHE","WUH","MU",null,-1,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc.addFC("WUH","CTU","MU",null,-1,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc.addFC("CTU","KMG","MU","YCT",1200,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
```

以上运价的例子均为假设，具体运价和使用条件须查询航空公司当时发布的运价，FC 的格式依航空公司发布运价时的规定写法而定。IBE 输入方式则可参照如上例子。

3.13. 预订成人缺口含运价 PNR

对于缺口程来说，实际上就相当于一个承运人为//或者/-的航段。我们还是按照简单加和的方式计算运价：

```
Nm: 1IBETESTER/IBEGROUP
SS:MU5102/Y/26JUN09/PEKSHA/NN1
SA: SHAHGH
SS:MU5131/Y/29DEC09/HGHPEK/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/25JUN09/pek099
SSR:FOID MU hk/ NI123456789098765432/p1
EI:不得签转/p1
```

```
TC:F/ARWNN/p1
FN:FCNY2280.00/SCNY2280.00/c0.00
-   /tCNY100.00CN/tExemPtYQ
-/p1
FC:PEK    MU  SHA  1130.00Y
-    // HGH
-    MU  PEK  1150.00Y
- CNY2280.00END
-/p1
FP:CASH,CNY/p1
@
  MU5102   Y  FR26JUN   PEKSHA  DK1   0800  1000
  MU5131   Y  TU29DEC   HGHPEK  DK1   0800  0955
XD42M  -  航空公司使用自动出票时限，请检查 PNR
```

输入了 FN/FC/FP/TC/EI。

产生的 PNR 结果：

```
 1.IBETESTER/IBEGROUP  XD42M
 2.  MU5102  Y   FR26JUN  PEKSHA  HK1   0800  1000          E  T2--
 3.    ARNK               SHAHGH
 4.  MU5131  Y   TU29DEC  HGHPEK  HK1   0800  0955      DACSE
 5.010-84669396
 6.PEK/T
 7.TL/1200/25JUN/PEK099
 8.FC/PEK  MU  SHA  1130.00Y  //HGH  MU  PEK  1150.00Y  CNY2280.00END/P1
 9.SSR  FOID  MU  HK1  NI123456789098765432/P1
10.SSR  ADTK  1E  BY  PEK23JUN09/0800  OR  CXL  MU5102  Y26JUN
11.RMK  CA/ZY484
12.FN/FCNY2280.00/SCNY2280.00/C0.00/XCNY100.00/TCNY100.00CN/TEXEMPTYQ/
    ACNY2380.00/P1
13.TC/ARWNN/P1
14.EI/BUDEQIANZHUAN 不得签转/P1
15.FP/CASH,CNY/P1
16.PEK099
```

IBE 生成这样的 PNR 的代码如后

```
        /*生成预定类对象*/
       SellSeat SellSeatexample= new SellSeat();

       /*添加旅客姓名*/
       String name="IBETESTER/IBEGROUP";
       SellSeatexample.addAdult(name);

       /*添加旅客乘坐航段信息*/
```

```
String airNo="MU5102";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-06-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*信息航段的添加*/
BookAirSeg arnkSeg;
arnkSeg = new BookAirSeg("SHA","HGH");
SellSeatexample.addAirSeg(arnkSeg);

/*添加旅客乘坐返程航段信息*/
String airNo2="MU5131";
char    fltClass2='Y';
String orgCity2="HGH";
String dstCity2="PEK";
String actionCode2="NN";
int tktNum2=1;
String departureTime2="2009-12-29";
SellSeatexample.addAirSeg(airNo2,fltClass2,orgCity2,dstCity2,actionCode2,tktNum2,departureTime2);





/*添加旅客身份证信息*/
String airline="MU";
String idtype="NI";
String id="123456789098765432";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);


/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-06-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*FN*/
```

```
BookFN fn = new BookFN();
//fn.setFn("FCNY2280.00/SCNY2280.00/C0.00/TCNY100.00CN/TEXEMPTYQ");
fn.setAmount(BookFN.F,"CNY",2280);
fn.setAmount(BookFN.S,"CNY",2280);
fn.setC(0);
fn.addTax(BookFN.T,"CNY",100,"CN");
fn.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"YQ");
fn.addPsgrname("IBETESTER/IBEGROUP");
SellSeatexample.addFN(fn);


/*FC*/
BookFC fc = new BookFC();
//fc.setFc("PEK MU SHA 1130.00Y // HGH MU PEK 1150.00Y CNY2280.00END");
fc.addFC("PEK","SHA","MU","Y",1130,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc.addFC("SHA","HGH","//",null,-1,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
//fc.addFCARNK();
fc.addFC("HGH","PEK","MU","Y",1150,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc.setMoneytype("CNY");
fc.addPsgrname("IBETESTER/IBEGROUP");
SellSeatexample.addFC(fc);

/*FP*/
BookFP fp = new BookFP("CASH,CNY","IBETESTER/IBEGROUP");
SellSeatexample.addFP(fp);

/*EI*/
BookEI ei = new BookEI("不得签转","");
ei.setPsgrName("IBETESTER/IBEGROUP");
SellSeatexample.addEI(ei);

/*TC*/
BookTC tc = new BookTC();
tc.setFreecode("ARWNN");
tc.setPsgrName("IBETESTER/IBEGROUP");
SellSeatexample.addTC(tc);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult   ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

如上代码，fn 格式没有变化，fc 按航段顺序，再 PEK-SHA 和 HGH-PEK 之间加上了一个

地面运输航段的 fc。该段 fc 也可以使用简单的语句 fc.addFCARNK()来实现。Fc 的运价计算中，航段必须按顺序加入。

3.14. 预订成人含不定期航段含运价 PNR

事实上，这种航段的 FC 与确定的航段是一样的。

```
Nm: 1IBETESTER/IBEGROUP
SS:MU5102/Y/26JUN09/PEKSHA/NN1
SNMU/ Y/ SHAPEK
CT:010-84669396
CT:pek/T
TK:TL/1200/25JUN09/pek099
SSR:FOID MU hk/ NI12345678909876543 2/p1
EI:不得签转
TC:F/ARWNN
FN:FCNY2260.00/SCNY2260.00/c0.00
-   /tCNY100.00CN/tExemPtYQ
FC:PEK    MU SHA 1130.00Y
-     MU PEK 1130.00Y
- CNY2260.00END
FP:CASH,CNY
@
  MU5102  Y FR26JUN  PEKSHA DK1   0800 1000
XD42Y -  航空公司使用自动出票时限，请检查 PNR
```

不定期航段，也需要按照航段的舱位和承运人按行程录入 FC 计算中，FN 中的价格也包括了不定期航段的价格。

产生的 PNR 结果：

```
 1.IBETESTER/IBEGROUP XD42Y
 2.  MU5102 Y   FR26JUN  PEKSHA HK1   0800 1000         E T2--
 3.  MUOPEN Y            SHAPEK
 4.010-84669396
 5.PEK/T
 6.TL/1200/25JUN/PEK099
 7.FC/PEK MU SHA 1130.00Y MU PEK 1130.00Y CNY2260.00END
 8.SSR FOID MU HK1 NI12345678909876543 2/P1
 9.SSR ADTK 1E BY PEK23JUN09/0800 OR CXL MU5102 Y26JUN
10.RMK CA/ZY49G
11.FN/FCNY2260.00/SCNY2260.00/C0.00/XCNY100.00/TCNY100.00CN/TEXEMPTYQ/
    ACNY2360.00
12.TC/ARWNN
```

```
13.EI/BUDEQIANZHUAN 不得签转
14.FP/CASH,CNY
15.PEK099
```

IBE 生成这样的 PNR 的代码如后

```
/*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加旅客乘坐航段信息*/
String airNo="MU5102";
char    fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=1;
String departureTime="2009-06-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);


/*添加旅客乘坐返程航段信息—为 OPEN 航段*/
String airNo2="MU";
char    fltClass2='Y';
String orgCity2="SHA";
String dstCity2="PEK";
SellSeatexample.addOpenAirSeg(airNo2,fltClass2,orgCity2,dstCity2);



/*添加旅客身份证信息*/
String airline="MU";
String idtype="NI";
String id="123456789098765432";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);


/*添加旅客联系组信息*/
String contactinfo="010-84669396";
```

```
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-06-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*FN*/
BookFN fn = new BookFN();
//fn.setFn("FCNY2260.00/SCNY2260.00/C0.00/TCNY100.00CN/TEXEMPTYQ");
fn.setAmount(BookFN.F,"CNY",2260);
fn.setAmount(BookFN.S,"CNY",2260);
fn.setC(0);
fn.addTax(BookFN.T,"CNY",100,"CN");
fn.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"YQ");
SellSeatexample.addFN(fn);

/*FC*/
BookFC fc = new BookFC();
//fc.setFc("PEK MU SHA 1130.00Y MU PEK 1130.00Y CNY2260.00END");
fc.addFC("PEK","SHA","MU","Y",1130,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc.addFC("SHA","PEK","MU","Y",1130,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);

fc.setMoneytype("CNY");
SellSeatexample.addFC(fc);




/*FP*/
BookFP fp = new BookFP("CASH,CNY","");
SellSeatexample.addFP(fp);

/*EI*/
BookEI ei = new BookEI("不得签转","");
SellSeatexample.addEI(ei);

/*TC*/
BookTC tc = new BookTC();
tc.setFreecode("ARWNN");
SellSeatexample.addTC(tc);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult   ssr=SellSeatexample.commit1();

/*PNR 结果*/
```

```
System.out.println(ssr.getPnrno());
```

3.15. 预订成人加儿童含运价 PNR

成人与儿童的运价不同，因此，FN 和 FC 必须分开，分别录入运价，在下面的例子中假设支付方式也不同，但是运价适用条件 EI 和旅行代码相同。

```
Nm: 1IBETESTER/IBEGROUP 1儿童旅客 CHD
SS:CA1501/Y/26DEC09/PEKSHA/NN2
SS:CA1502/Y/29DEC09/SHAPEK/NN2
CT:010-84669396
CT:pek/T
TK:TL/1200/25DEC09/pek099
SSR:FOID CA hk/ NI123456789/p1
SSR:FOID CA hk/ NI234567890/p2
EI:不得签转
TC:F/ARWNN
FN:FCNY1140.00/SCNY1140.00/c0.00
-   /tExemPtCN/tExemPtYQ
-/p2
FC:PEK    CA SHA 570.00YCH
-     CA PEK 570.00YCH
- CNY1140.00END
-/p2
FN:FCNY2260.00/SCNY2260.00/c0.00
-   /tCNY100.00CN/tExemPtYQ
FC:PEK    CA SHA 1130.00Y
-     CA PEK 1130.00Y
- CNY2260.00END
FP:CASH,CNY/p1
FP:CC/p2
@
XD432 -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA1501  Y SA26DEC  PEKSHA DK2   0830 1035
  CA1502  Y TU29DEC  SHAPEK DK2   1150 1415
 □航空公司使用自动出票时限，请检查 PNR□
```

产生的 PNR 结果：

```
  1.儿童旅客 CHD 2.IBETESTER/IBEGROUP XD432
  3.  CA1501 Y   SA26DEC  PEKSHA HK2   0830 1035         E T3--
  4.  CA1502 Y   TU29DEC  SHAPEK HK2   1150 1415         E --T3
  5.010-84669396
  6.PEK/T
  7.TL/1200/25DEC/PEK099
```

```
 8.FC/PEK CA SHA 570.00YCH CA PEK 570.00YCH CNY1140.00END/P1
 9.FC/PEK CA SHA 1130.00Y CA PEK 1130.00Y CNY2260.00END
10.SSR FOID CA HK1 NI234567890/P1
11.SSR FOID CA HK1 NI123456789/P2
12.SSR ADTK 1E BY PEK23DEC09/0830 OR CXL CA1501 Y26DEC
13.RMK CA/ZY4B2
14.FN/FCNY1140.00/SCNY1140.00/C0.00/TEXEMPTCN/TEXEMPTYQ/ACNY1140.00/P1
15.FN/FCNY2260.00/SCNY2260.00/C0.00/XCNY100.00/TCNY100.00CN/TEXEMPTYQ/
    ACNY2360.00
16.TC/ARWNN
17.EI/BUDEQIANZHUAN 不得签转
18.FP/CASH,CNY/P2
19.FP/CC/P1                                                         -
20.PEK099
```

IBE 生成这样的 PNR 的代码如后

```
 /*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
String name="IBETESTER/IBEGROUP";
SellSeatexample.addAdult(name);

/*添加旅客姓名*/
String name_c="儿童旅客";
SellSeatexample.addChild(name_c);

/*添加旅客乘坐航段信息*/
String airNo="CA1501";
char   fltClass='Y';
String orgCity="PEK";
String dstCity="SHA";
String actionCode="NN";
int tktNum=2;
String departureTime="2009-12-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客乘坐航段信息*/
String airNo2="CA1502";
char   fltClass2='Y';
String orgCity2="SHA";
String dstCity2="PEK";
String actionCode2="NN";
int tktNum2=2;
String departureTime2="2009-12-29";
```

```
SellSeatexample.addAirSeg(airNo2,fltClass2,orgCity2,dstCity2,actionCode2,tktNum2,departureTime2);



/*添加旅客身份证信息*/
String airline="CA";
String idtype="NI";
String id="123456789";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);

/*添加旅客身份证信息*/
String airline_c="CA";
String idtype_c="NI";
String id_c="234567890";
SellSeatexample.addSSR_FOID(airline_c,idtype_c,id_c,name_c);

/*添加旅客联系组信息*/
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-12-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*FN*/
BookFN fn = new BookFN();
//fn.setFn("FCNY2260.00/SCNY2260.00/C0.00/TCNY100.00CN/TEXEMPTYQ");
fn.setAmount(BookFN.F,"CNY",2260);
fn.setAmount(BookFN.S,"CNY",2260);
fn.setC(0);
fn.addTax(BookFN.T,"CNY",100,"CN");
fn.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"YQ");
SellSeatexample.addFN(fn);

/*FC*/
BookFC fc = new BookFC();
//fc.setFc("PEK MU SHA 1130.00Y MU PEK 1130.00Y CNY2260.00END");
fc.addFC("PEK","SHA","CA","Y",1130,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc.addFC("SHA","PEK","CA","Y",1130,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);

fc.setMoneytype("CNY");
SellSeatexample.addFC(fc);

/*FN*/
```

```
BookFN fn_c = new BookFN();
//fn_c.setFn("FCNY1140.00/SCNY1140.00/C0.00/TEXEMPTCN/TEXEMPTYQ");
fn_c.setAmount(BookFN.F,"CNY",1140);
fn_c.setAmount(BookFN.S,"CNY",1140);
fn_c.setC(0);
fn_c.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"CN");
fn_c.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"YQ");
fn_c.addPsgrname(name_c);
SellSeatexample.addFN(fn_c);

/*FC*/
BookFC fc_c = new BookFC();
//fc_c.setFc("PEK MU SHA 570.00Y MU PEK 570.00Y CNY1140.00END");
fc_c.addFC("PEK","SHA","CA","YCH",570,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc_c.addFC("SHA","PEK","CA","YCH",570,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);

fc_c.setMoneytype("CNY");
fc_c.addPsgrname(name_c);
SellSeatexample.addFC(fc_c);

/*FP*/
BookFP fp = new BookFP("CASH,CNY",name);
SellSeatexample.addFP(fp);
/*FP*/
BookFP fp_c = new BookFP("CC",name_c);
SellSeatexample.addFP(fp_c);

/*EI*/
BookEI ei = new BookEI("不得签转","");
SellSeatexample.addEI(ei);

/*TC*/
BookTC tc = new BookTC();
tc.setFreecode("ARWNN");
SellSeatexample.addTC(tc);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
/*EI*/
BookEI ei = new BookEI("不得签转","");
SellSeatexample.addEI(ei);
```

```
/*TC*/
BookTC tc = new BookTC();
tc.setFreecode("ARWNN");
SellSeatexample.addTC(tc);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

在 fn_c 和 fc_c 后面分别调用了各自的方法 addPSgrname(String),参数表示使用该 FN/FC 的旅客的姓名，需要与订座时输入的旅客姓名完全一致。当表示某条 FN/FC 不是适用于 PNR 中所有旅客，而仅适用于某一位/几位旅客时，可以调用此方法。而没有录入姓名的 FN/FC 表示该 FN/FC 适用于 PNR 中所有没有特别指定 FN/FC 的旅客。FN 和 FC 可以指定多个旅客适用该 FN/FC。

FP 在构造时指定了旅客姓名，表示该 FP 只适用于指定姓名的旅客。FP 只能指定一位旅客适用该 FP。

TC，EI 也各自调用了其 setPsgrName 方法，也表示此 TC/EI 只适用于指定姓名的旅客。TC/EI 只能指定一个旅客适用该 TC/EI。

录入 FN/FC/FP 时，应当保证每位旅客都有匹配上的 FN/FC/FP，并且不能为同一位旅客指定两个或以上的 FN/FC/FP。如果舱位要求输入 EI/TC，则也必须满足每位旅客都能匹配上 EI/TC，且不能被指定多个 EI/TC。

以 FN 为例

```
 /*生成预定类对象*/
SellSeat SellSeatexample= new SellSeat();

/*添加旅客姓名*/
SellSeatexample.addAdult(“IBE/A”);
SellSeatexample.addAdult(“IBE/B”);
SellSeatexample.addAdult(“IBE/C”);
……

BookFN fn1=new BookFN();
BookFN fn2 = new BookFN();

fn1.addPsgrname(“IBE/A”);


SellSeatexample.addFN(fn1);
SellSeatexample.addFN(fn2);
```

如上代码，pnr 中旅客为 IBE/A,IBE/B,IBE/C 三位，fn1 设定了旅客姓名为 IBE/A，

而 fn2 没有设定姓名，则表示 fn2 匹配的旅客为除 IBE/A 之外的旅客，这样的输入是合法的。

```
BookFN fn1=new BookFN();
BookFN fn2 = new BookFN();

fn1.addPsgrname("IBE/A");
fn2.addPsgrname("IBE/B");
fn2.addPsgrname("IBE/C");

SellSeatexample.addFN(fn1);
SellSeatexample.addFN(fn2);
```

如上，fn1 设定旅客姓名为 IBE/A，fn2 设定旅客为 IBE/B 和 IBE/C。这样也是合法的。

```
BookFN fn1=new BookFN();
BookFN fn2 = new BookFN();

fn1.addPsgrname("IBE/A");
fn1.addPsgrname("IBE/B");
fn2.addPsgrname("IBE/B");
fn2.addPsgrname("IBE/C");

SellSeatexample.addFN(fn1);
SellSeatexample.addFN(fn2);
```

如上，fn1 设定旅客姓名为 IBE/A 和 IBE/B，fn2 设定旅客为 IBE/B 和 IBE/C。这样 IBE/B 同时被匹配上两条 FN，这样是不满足输入条件的。

```
BookFN fn1=new BookFN();
BookFN fn2 = new BookFN();
BookFN fn = new BookFN();

fn1.addPsgrname("IBE/A");
fn2.addPsgrname("IBE/B");
fn2.addPsgrname("IBE/C");

SellSeatexample.addFN(fn1);
SellSeatexample.addFN(fn2);
SellSeatexample.addFN(fn);
```

如上，fn1 设定旅客姓名为 IBE/A fn2 设定旅客为 IBE/B 和 IBE/C。这样 fn 匹配不上任何一个旅客，这样也是不满足输入条件的。

FC 也类似。FP，TC，EI 的对象只能指定一个旅客姓名，除此之外，需要满足的约束条件

与 FN 一致。

3.16. 成人加婴儿含运价的 PNR

婴儿必须单独为其输入 FN/FC/FP 项。但是，由于婴儿是不占座旅客，因此婴儿 FC，FN，FP 如果需要添加姓名指定适用于该旅客所携带的婴儿，则必须添加携带婴儿的成人旅客的姓名。婴儿的 FC/FN/FP 均应当指定其 infant 属性为 true。此例中仍然假定成人与婴儿的支付方式不同。

```
Nm: 1IBETESTER/IBEGROUP
SS:CA1501/Y/26DEC09/PEKSHA/NN1
SS:CA1502/Y/29DEC09/SHAPEK/NN1
CT:010-84669396
CT:pek/T
TK:TL/1200/25DEC09/pek099
XN:IN/王婴儿 INF(JAN09)/p1
SSR:FOID CA hk/ NI123456789/p1
SSR:INFt CA nn1 PEKSHA 1501 Y 26DEC09 WANG/YINGER 03JAN09/p1
SSR:INFt CA nn1 SHAPEK 1502 Y 29DEC09 WANG/YINGER 03JAN09/p1
EI:不得签转
TC:F/ARWNN
FN:IN/FCNY220.00/SCNY220.00/c0.00
-   /tExemPtCN/tExemPtYQ
-/p1
FC:IN/PEK    CA SHA 110.00YIN
-     CA PEK 110.00YIN
- CNY220.00END
-/p1
FN:FCNY2260.00/SCNY2260.00/c0.00
-   /tCNY100.00CN/tExemPtYQ
FC:PEK    CA SHA 1130.00Y
-     CA PEK 1130.00Y
- CNY2260.00END
FP:CASH,CNY/p1
FP:IN/CC/p1
@
XD43D -EOT SUCCESSFUL, BUT ASR UNUSED FOR 1 OR MORE SEGMENTS
  CA1501  Y SA26DEC  PEKSHA DK1   0830 1035
  CA1502  Y TU29DEC  SHAPEK DK1   1150 1415
 □航空公司使用自动出票时限，请检查 PNR□
```

产生的 PNR 结果：婴儿的 FN/FC/FP 前均有 IN 标识

```
 1.IBETESTER/IBEGROUP XD43D
 2.  CA1501 Y   SA26DEC  PEKSHA HK1   0830 1035         E T3--
 3.  CA1502 Y   TU29DEC  SHAPEK HK1   1150 1415         E --T3
 4.010-84669396
 5.PEK/T
 6.TL/1200/25DEC/PEK099
 7.FC/IN/PEK CA SHA 110.00YIN CA PEK 110.00YIN CNY220.00END/P1
 8.FC/PEK CA SHA 1130.00Y CA PEK 1130.00Y CNY2260.00END
 9.SSR FOID CA HK1 NI123456789/P1
10.SSR ADTK 1E BY PEK23DEC09/0830 OR CXL CA1501 Y26DEC
11.SSR INFT CA □KK1□PEKSHA 1501 Y26DEC WANG/YINGER 03JAN09/P1
12.SSR INFT CA □KK1□SHAPEK 1502 Y29DEC WANG/YINGER 03JAN09/P1
13.OSI YY 1INF WANGYINGER/P1
14.RMK CA/ZY4BX
15.FN/IN/FCNY220.00/SCNY220.00/C0.00/TEXEMPTCN/TEXEMPTYQ/ACNY220.00/P1
16.FN/FCNY2260.00/SCNY2260.00/C0.00/XCNY100.00/TCNY100.00CN/TEXEMPTYQ/
    ACNY2360.00
17.TC/ARWNN
18.EI/BUDEQIANZHUAN 不得签转
19.FP/CASH,CNY/P1
20.FP/IN/CC/P1
21.XN/IN/王婴儿 INF(JAN09)/P1
22.PEK099
```

IBE 生成这样的 PNR 的代码如后

```
        /*生成预定类对象*/
    SellSeat SellSeatexample= new SellSeat();

    /*添加旅客姓名*/
    String name="IBETESTER/IBEGROUP";
    SellSeatexample.addAdult(name);

    /*添加婴儿的基本信息*/
    String infant_name = "王婴儿";
    String infant_birth = "2009-01-03";
    SellSeatexample.addInfant(infant_birth,name,infant_name);



    /*添加旅客乘坐航段信息*/
    String airNo="CA1501";
    char   fltClass='Y';
    String orgCity="PEK";
    String dstCity="SHA";
    String actionCode="NN";
```

```
int tktNum=1;
String departureTime="2009-12-26";
SellSeatexample.addAirSeg(airNo,fltClass,orgCity,dstCity,actionCode,tktNum,departureTime);

/*添加旅客乘坐航段信息*/
String airNo2="CA1502";
char   fltClass2='Y';
String orgCity2="SHA";
String dstCity2="PEK";
String actionCode2="NN";
int tktNum2=1;
String departureTime2="2009-12-29";
SellSeatexample.addAirSeg(airNo2,fltClass2,orgCity2,dstCity2,actionCode2,tktNum2,departureTime2);



/*添加旅客身份证信息*/
String airline="CA";
String idtype="NI";
String id="123456789";
SellSeatexample.addSSR_FOID(airline,idtype,id,name);

/*为婴儿的第 0 航段(即去程)添加 SSR INFT */
String airline0_i = "CA";
String segment0_i = "PEKSHA";
String fltNo0_i = "1501";
String fltclass0_i = "Y";
String date0_i = "26DEC09";
String infantName = "WANG/YINGER";
String infantBirth = "03JAN09";
String freeText = "";
SellSeatexample.addSSR_INFT(airline0_i, segment0_i, fltNo0_i, fltclass0_i,
date0_i, infantName, infantBirth, freeText, name);

/*为婴儿的第 1 航段(即回程)添加 SSR INFT */
String airline1_i = "CA";
String segment1_i = "SHAPEK";
String fltNo1_i = "1502";
String fltclass1_i = "Y";
String date1_i = "29DEC09";
SellSeatexample.addSSR_INFT(airline1_i, segment1_i, fltNo1_i, fltclass1_i,
        date1_i, infantName, infantBirth, freeText, name);

/*添加旅客联系组信息*/
```

```
String contactinfo="010-84669396";
SellSeatexample.addContact(contactinfo);

/*添加旅客出票时限*/
String dateLimit="2009-12-25 12:00:00";
SellSeatexample.setTimelimit(dateLimit);

/*FN*/
BookFN fn = new BookFN();
//fn.setFn("FCNY2260.00/SCNY2260.00/C0.00/TCNY100.00CN/TEXEMPTYQ");
fn.setAmount(BookFN.F,"CNY",2260);
fn.setAmount(BookFN.S,"CNY",2260);
fn.setC(0);
fn.addTax(BookFN.T,"CNY",100,"CN");
fn.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"YQ");
SellSeatexample.addFN(fn);

/*FC*/
BookFC fc = new BookFC();
//fc.setFc("PEK MU SHA 1130.00Y MU PEK 1130.00Y CNY2260.00END");
fc.addFC("PEK","SHA","CA","Y",1130,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc.addFC("SHA","PEK","CA","Y",1130,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);

fc.setMoneytype("CNY");
SellSeatexample.addFC(fc);

/*婴儿 FN*/
BookFN fn_i = new BookFN();
//fn_i.setFn("FCNY220.00/SCNY220.00/C0.00/TEXEMPTCN/TEXEMPTYQ");
fn_i.setAmount(BookFN.F,"CNY",220);
fn_i.setAmount(BookFN.S,"CNY",220);
fn_i.setC(0);
fn_i.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"CN");
fn_i.addTax(BookFN.T,"",BookFN.EXEMPTTAX,"YQ");
fn_i.addPsgrname(name);

  fn_i.setInfant(true);
SellSeatexample.addFN(fn_i);

/*婴儿 FC*/
BookFC fc_i = new BookFC();
//fc_c.setFc("PEK MU SHA 110.00Y MU PEK 110.00Y CNY220.00END");
fc_i.addFC("PEK","SHA","CA","YIN",110,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
fc_i.addFC("SHA","PEK","CA","YIN",110,-1,-1,null,null,false,true,null,-1,null,-1,null,null,null);
```

```
fc_i.setMoneytype("CNY");
fc_i.addPsgrname(name);
fc_i.setInfant(true);
SellSeatexample.addFC(fc_i);

/*FP*/
BookFP fp = new BookFP("CASH,CNY",name);
SellSeatexample.addFP(fp);
/*FP*/
BookFP fp_i = new BookFP("CC",name);
fp_i.setInfant(true);
SellSeatexample.addFP(fp_i);

/*EI*/
BookEI ei = new BookEI("不得签转","");
SellSeatexample.addEI(ei);

/*TC*/
BookTC tc = new BookTC();
tc.setFreecode("ARWNN");
SellSeatexample.addTC(tc);

/*完成 PNR 必需信息输入，递交主机，生成 PNR*/
SSResult    ssr=SellSeatexample.commit1();

/*PNR 结果*/
System.out.println(ssr.getPnrno());
```

订座常见错误信息和异常

出错信息和可能的原因提示:

SSUMAGEException	无人陪伴旅客的年龄
SSFltUnableException	航段已经不能订取
SSFltNumberException	指定的航班号不存在或有错误
SSSegmentContinuityException	航段组不连续
SSAuthorityException	无权订取航段或者无权按指定的行动代码订取航段
SSNameLengthException	姓名长度不正确，或者姓长度不够，或者需要在姓和名之间用“/”分开
SSNameMissingException	散客订座，但是没有输入任何姓名

SSContactMissingException	没有输入任何联系组
SSTicketMissingException	没有输入任何出票状态组（出票时限）
SSSegmentMissingException	没有输入任何航段
SSTktDateException	设置的出票时限不正确，早于订座生成PNR 的时间或者晚于第一航段起飞时间
SSInvalidCharException	姓名、联系组、SSR 信息、OSI 信息、RMK 信息，婴儿姓名或者证件号等录入信息中有不合法字符
SSInvalidInfoException	输入的信息格式不正确，如 SSR FOID 里的证件信息，RMK MP 里的手机号
SSFlightEnrouteException	要订取的航班已经起飞
SSActionCodeException	订取的航段或者 SSR 里使用的行动代码不正确,SSR 可以参考各 SSR 代码的输入规范。航段订取根据舱位代码而定。
SSOfficeException	在出票时限等地输入了错误的 office 代码。
SSAirlineException	SSR/OSI 中录入的航空公司代码不正确

4. PNR 提取

实现类：com.travelsky.ibe.client.pnr.RT
com.travelsky.ibe.client.pnr.PNRAirSeg
com.travelsky.ibe.client.pnr.PNRBA
com.travelsky.ibe.client.pnr.PNRContact
com.travelsky.ibe.client.pnr.PNREI
com.travelsky.ibe.client.pnr.PNREnvelop
com.travelsky.ibe.client.pnr.PNRFC
com.travelsky.ibe.client.pnr.PNRFN
com.travelsky.ibe.client.pnr.PNRFP
com.travelsky.ibe.client.pnr.PNRInfant
com.travelsky.ibe.client.pnr.PNROI
com.travelsky.ibe.client.pnr.PNROSI
com.travelsky.ibe.client.pnr.PNROther
com.travelsky.ibe.client.pnr.PNRPassenger
com.travelsky.ibe.client.pnr.PNRResp
com.travelsky.ibe.client.pnr.PNRRMK
com.travelsky.ibe.client.pnr. PNRSSR_DOCS
com.travelsky.ibe.client.pnr. PNRSSR_FOID
com.travelsky.ibe.client.pnr. PNRSSR_FQTV
com.travelsky.ibe.client.pnr. PNRSSR_PSPT
com.travelsky.ibe.client.pnr. PNRSSR_RQST
com.travelsky.ibe.client.pnr. PNRSSR_SEAT
com.travelsky.ibe.client.pnr. PNRSSR_TKNE
com.travelsky.ibe.client.pnr.PNRSSR
com.travelsky.ibe.client.pnr.PNRTC
com.travelsky.ibe.client.pnr.PNRTkt
com.travelsky.ibe.client.pnr002EPNRTktNo

4.1. 根据 PNR 编号提取 PNR

➢ eTerm RT 查询过程

eTerm 中运用主机指令提取 PNR：

```
RT:RWGW2       --①

**ELECTRONIC TICKET PNR**
 1.韩杨 RWGW2
 2.  CA1501 G   SU02AUG  PEKSHA RR1   0830 1040          E
 3.SHA/T SHA/T 021-61327000/JEBSEN AIR SERVICE CO.,LTD/ZHOU ZHI YANG ABCDEFG
 4.ZHY FOR NOVOZYMES-BJ-T
 5.T
 6.SSR FOID CA HK1 NI****************/P1
 7.SSR OTHS CA 3CN315
 8.SSR ADTK 1E BY SHA23JUL09/1448 OR CXL CA1501 G02AUG
 9.SSR TKNE CA HK1 PEKSHA 1501 G02AUG 99996****1194/1/P1
10.OSI CA CTCT 158012828

pn1
11.RMK MP 15801282879/P1                                          -
12.RMK BUKUSN 1116635/P1
13.RMK CA/EMS5T
14.RMK AUTOMATIC FARE QUOTE
15.FN/A/FCNY680.00/SCNY680.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY730.
00
16.TN/999-96****1194/P1
17.FP/CASH,CNY/*3CN315
18.SHA178
```

提取 PNR(RT)：IBE 可对应类 RT 的 retrieve(String pnrno)方法。

pnrno：pnr 编号，如上图：BSQXP

返回结果为 RTResult

➢ IBE 实现 PNR 提取

```
/*
 * 实现 PNR 信息提取
 * */
/*生成 RT 对象*/
  RT   rtExample=new RT();

  /*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("RWGW2”);

  /*显示提取结果*/
  System.out.println(rtres);
```

RT 显示结果

```
PNRNO:RWGW2

团体票标识:散客票

姓名组
1.1.韩杨    成人    韩杨

航段组
1. 2. CA1501 G PEKSHA RR1 Sun Aug 02 08:30:00 CST 2009 Sun Aug 02 10:40:00 CST 2009 电子票
离港航站楼:--    到港航站楼:--    普通航段
CA1501 G     SU02AUG    PEKSHA RR1     0830 1040              E

联系组
1.3.SHA SHA/T SHA/T 021-61327000/JEBSEN AIR SERVICE CO.,LTD/ZHOU ZHI YANG ABCDEFG
2.4. ZHY FOR NOVOZYMES-BJ-T

开账地址组（BA）

出票组
1.5.T null    已出票     备注:T
票号组
1.16.999-9652101194 P1

备注组
1.11.MP 15801282879/P1 P1
2.12.BUKUSN 1116635/P1 P1
3.13.CA/EMS5T
4.14.AUTOMATIC FARE QUOTE

特别服务组
1.6.SSR    FOID    CA    HK 1    NI ****************** P1
2.7.SSR OTHS CA 3CN315<-->7.OTHS|CA||
3.8.SSR ADTK 1E BY SHA23JUL09/1448 OR CXL CA1501 G02AUG<-->8.ADTK|1E||
4.9.9.SSR    CA HK PEKSHA 1501 G 02AUG 9999****01194 1 P1    普通票 P1

其他服务组
1.10.CA CTCT 15801282879/P1 P1

婴儿组

其他
1.15.FN/A/FCNY680.00/SCNY680.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY730.00
2.17.FP/CASH,CNY/*3CN315
```

FN 组
1.原文：FN/A/FCNY680.00/SCNY680.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY730.00
票面价：CNY680.00
实收：CNY680.00
代理费率：3.00
附加费用之和：CNY50.00
税项：　　CN:CNY50.00　　YQ:EXEMPT
总价：CNY730.00

FC 组

FP 组
*1.17.FP CASH,CNY　原文:FP/CASH,CNY/*3CN315*

OI 组

TC 组

EI 组

责任组
1.18.　SHA178

TIPS：

IBE 提取 PNR 信息中，可以用 retrieveCompletely 方法获取整个 PNR 的信息，包括历史部分的全文。

出错信息提示:
- RTPNRCancelledException　　PNR 已经删除。
- RTNoPNRException　　PNR 不存在。

4.2. 基本 PNR 信息

IBE 提取 PNR 信息后得到一个结构化的结果 RTResult，我们通过对这个 RTResult 进行解析，获取 PNR 中的旅客信息

4.3. 提取旅客组

eTerm 中的旅客组信息：

```
RT:SK43X

1.IBETESTER/IBEGROUP 2.XIAO/LIN CHD SK43X            --① ②
 3.  CA1501 Y   SU02AUG  PEKSHA HK2   0830 1040         E T3--
 4.TRAVELSKY/010-84669396
 5.TL/1200/01AUG/PEK099
 6.SSR FOID CA HK1 NI123456789/P1
 7.SSR ADTK 1E BY PEK30JUL09/0830 OR CXL CA1501 Y02AUG
 8.SSR INFT CA □KK1□PEKSHA 1501 Y02AUG WANG/YINGER 03JAN09/P1
 9.OSI YY 1INF WANGYINGER/P1
10.RMK CA/ZQ428
11.XN/IN/王婴儿 INF(JAN09)/P1                            --③
12.PEK099
```

➢ IBE 实现提取旅客信息

```
/*
 * 实现 PNR 旅客信息提取
 * */
/*生成 RT 对象*/
   RT   rtExample=new RT();

   /*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("SK43X”);



 /*显示旅客信息*/
 System.out.println("显示成人和儿童旅客");
 for(int i=0;i<rtres.getPassengersCount();i++){
      PNRPassenger pnrPass=rtres.getPassengerAt(i);
       System.out.print("PNR 序号:"+pnrPass.getIndex());
       System.out.print("         旅客姓名:"+pnrPass.getName());
       System.out.print("           年龄:"+pnrPass.getAge());
       System.out.print("   PNR 中姓名:"+pnrPass.getNameInPnr());
       System.out.println();
     }

   System.out.println();

    System.out.println("显示婴儿旅客");
   for(int i=0;i<rtres.getInfantsCount();i++){
         PNRInfant pnrInfant=rtres.getInfantAt(i);
          System.out.print("PNR 序号:"+pnrInfant.getIndex());
          System.out.print("         旅客姓名:"+pnrInfant.getName());
```

```
System.out.print("    同机携带者:"+pnrInfant.getCarrier());
System.out.print("    PNR 中姓名:"+pnrInfant.getNameInPnr());
System.out.print("    生日:"+pnrInfant.getBrithString());
System.out.println();
}
```

PNR 旅客信息查询显示结果

```
显示成人和儿童旅客信息
PNR 序号:1        旅客姓名:IBETESTER/IBEGROUP        年龄:0    PNR 中姓名:IBETESTER/IBEGROUP
PNR 序号:2        旅客姓名:XIAO/LIN        年龄:0    PNR 中姓名:XIAO/LIN CHD

显示婴儿旅客信息
PNR 序号:11        旅客姓名:王婴儿    同机携带者:P1    PNR 中姓名:王婴儿 INF(JAN09)    生日:Thu Jan 01
00:00:00 CST 2009
```

4.4. 提取航段组

eTerm 中的航段组信息：

```
RT:SK49Z

 1.IBETESTER/IBEGROUP SK49Z
 2.  CA1501 Y   SA26DEC  PEKSHA HK1   0830 1035          E T3--        --①
 3.  CA1522 Y   TU29DEC  SHAPEK HK1   1750 2020          E --T3        --②
 4.TRAVELSKY/010-84669396
 5.TL/1200/25DEC/PEK099
 6.SSR FOID CA HK1 NI123456789/P1
 7.SSR ADTK 1E BY PEK23DEC09/0830 OR CXL CA1501 Y26DEC
 8.RMK CA/KQ06B
 9.PEK099
```

- IBE 实现提取航段信息

```
/*
 * 实现 PNR 旅客信息提取
 * */
/*生成 RT 对象*/
   RT   rtExample=new RT();

   /*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("SK49Z”);



 /*显示航段信息*/
System.out.println("显示航段组信息:");
    for(int i=0;i<rtres.getAirSegsCount();i++){
        PNRAirSeg pnrAirseg=rtres.getAirSegAt(i);
        System.out.println("PNR 序号:"+pnrAirseg.getIndex());
        System.out.println("    航班号:"+pnrAirseg.getAirNo());
        System.out.println("    出发城市"+pnrAirseg.getOrgCity());
        System.out.println("    到达城市:"+pnrAirseg.getDesCity());
        System.out.println("    出发时间"+pnrAirseg.getDepartureTimeString());
        System.out.println("    到达时间:"+pnrAirseg.getArrivalTimeString());
        System.out.println("    航班舱位:"+pnrAirseg.getFltClass());
       }
```

PNR 航段组信息查询显示结果

```
显示航段组信息:
            PNR 序号:2
             航班号:CA1501
             出发城市 PEK
             到达城市:SHA
             出发时间 Sat Dec 26 08:30:00 CST 2009
             到达时间:Sat Dec 26 10:35:00 CST 2009
             航班舱位:Y
            PNR 序号:3
             航班号:CA1522
             出发城市 SHA
             到达城市:PEK
             出发时间 Tue Dec 29 17:50:00 CST 2009
             到达时间:Tue Dec 29 20:20:00 CST 2009
             航班舱位:Y
```

4.5. 提取联系组

eTerm 中的联系组信息：

```
RT:SK49Z

 1.IBETESTER/IBEGROUP SK49Z
 2.  CA1501 Y   SA26DEC  PEKSHA HK1   0830 1035          E T3--
 3.  CA1522 Y   TU29DEC  SHAPEK HK1   1750 2020          E --T3
 4.TRAVELSKY/010-84669396                                           --①
 5.TL/1200/25DEC/PEK099
 6.SSR FOID CA HK1 NI123456789/P1
 7.SSR ADTK 1E BY PEK23DEC09/0830 OR CXL CA1501 Y26DEC
 8.RMK CA/KQ06B
 9.PEK099
```

- IBE 实现提取联系组信息

```
/*
 * 实现 PNR 旅客信息提取
 * */
/*生成 RT 对象*/
   RT   rtExample=new RT();

   /*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("SK49Z");


 /*显示联系组信息*/
System.out.println("显示联系组信息:");
   for(int i=0;i<rtres.getContactsCount();i++){
       PNRContact pnrContact=(PNRContact)rtres.getContactAt(i);
        System.out.print("PNR 序号:"+pnrContact.getIndex());
   }
```

PNR 联系组信息查询显示结果

```
          显示联系组信息:
PNR 序号:4      联系组信息:TRAVELSKY/010-84669396
```

4.6. 提取出票组

eTerm 中的出票组信息：

```
RT:RWGW2

**ELECTRONIC TICKET PNR**
 1.韩杨 RWGW2
 2.  CA1501 G   SU02AUG  PEKSHA RR1   0830 1040          E
 3.SHA/T SHA/T 021-61327000/JEBSEN AIR SERVICE CO.,LTD/ZHOU ZHI YANG ABCDEFG
 4.ZHY FOR NOVOZYMES-BJ-T
 5.T
 6.SSR FOID CA HK1 NI****************/P1
 7.SSR OTHS CA 3CN315
 8.SSR ADTK 1E BY SHA23JUL09/1448 OR CXL CA1501 G02AUG
 9.SSR TKNE CA HK1 PEKSHA 1501 G02AUG 99996****1194/1/P1
10.OSI CA CTCT 158012828

pn1
11.RMK MP 15801282879/P1                                                   -
12.RMK BUKUSN 1116635/P1
13.RMK CA/EMS5T
14.RMK AUTOMATIC FARE QUOTE
15.FN/A/FCNY680.00/SCNY680.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY730.
00
16.TN/999-96****1194/P1
17.FP/CASH,CNY/*3CN315
18.SHA178                                                              --①
```

➢ IBE 实现提取出票组信息

```
/*
 * 实现 PNRREMARK 信息提取
 * */
/*生成 RT 对象*/
  RT   rtExample=new RT();

  /*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("SK49Z");


/*显示出票组信息组*/
 System.out.println("显示出票组信息");
 for(int i=0;i<rtres.getTktsCount();i++){
      PNRTkt pnrTkt=(PNRTkt)rtres.getTktAt(i);
      System.out.print("PNR 序号:"+pnrTkt.getIndex());
     System.out.print("出票的 OFFICE:"+pnrTkt.getOffice());
 }
```

PNR 出票组信息查询显示结果

```
显示出票组信息
PNR 序号:5
出票的 OFFICE:
```

4.7. 提取特殊服务组

eTerm 中的特殊服务组信息：

```
RT:RWGW2      --①

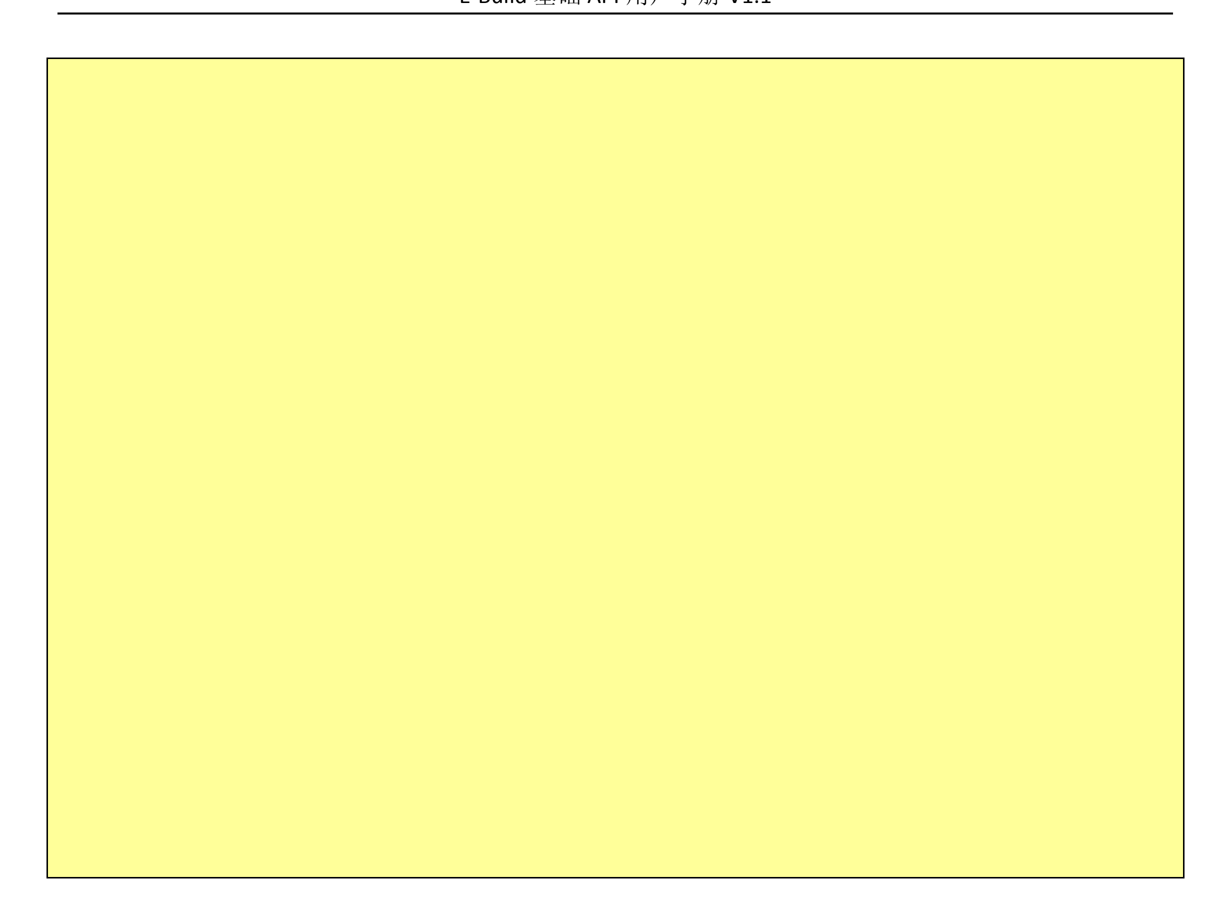
**ELECTRONIC TICKET PNR**
 1.韩杨 RWGW2
 2.  CA1501 G   SU02AUG  PEKSHA RR1   0830 1040          E
 3.SHA/T SHA/T 021-61327000/JEBSEN AIR SERVICE CO.,LTD/ZHOU ZHI YANG ABCDEFG
 4.ZHY FOR NOVOZYMES-BJ-T
 5.T
 6.SSR FOID CA HK1 NI****************/P1                      --①
 7.SSR OTHS CA 3CN315                                         --②
 8.SSR ADTK 1E BY SHA23JUL09/1448 OR CXL CA1501 G02AUG        --③
 9.SSR TKNE CA HK1 PEKSHA 1501 G02AUG 99996****1194/1/P1      --④
10.OSI CA CTCT 158012828
```

- IBE 实现提取特殊服务组信息

```
/*
 * 实现 PNR 旅客信息提取
 * */
/*生成 RT 对象*/
RT   rtExample=new RT();

/*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("RWGW2”);


/*显示特殊服务组信息*/
System.out.println("显示特殊服务组信息:");
for(int i=0;i<rtres.getSSRsCount();i++){
        PNRSSR pnrSsr=(PNRSSR)rtres.getSSRAt(i);
        System.out.print("PNR 序号:"+pnrSsr.getIndex());
        System.out.print("    服务类型"+pnrSsr.getSSRType());
        System.out.print("    服务信息"+pnrSsr.getServeInfo());
        System.out.println("    旅客标志"+pnrSsr.getPsgrID());
}
```



PNR 特殊服务组信息查询显示结果

显示特殊服务组信息:
*PNR 序号:6　服务类型 FOID　服务信息 FOID CA HK1 NI**************4/P1　旅客标志 P1*
PNR 序号:7　服务类型 OTHS　服务信息 OTHS CA 3CN315　旅客标志
PNR 序号:8　服务类型 ADTK　服务信息 ADTK 1E BY SHA23JUL09/1448 OR CXL CA1501 G02AUG　旅客标志
*PNR 序号:9　服务类型 TKNE　服务信息 TKNE CA HK1 PEKSHA 1501 G02AUG 9999*******94/1/P1　旅客标志 P1*

4.8. 提取 OSI

eTerm 中的 OSI 信息：

```
RT:RWGW2      --①

**ELECTRONIC TICKET PNR**
 1.韩杨 RWGW2
 2.  CA1501 G   SU02AUG  PEKSHA RR1   0830 1040          E
 3.SHA/T SHA/T 021-61327000/JEBSEN AIR SERVICE CO.,LTD/ZHOU ZHI YANG ABCDEFG
 4.ZHY FOR NOVOZYMES-BJ-T
 5.T
 6.SSR FOID CA HK1 NI****************/P1
 7.SSR OTHS CA 3CN315
 8.SSR ADTK 1E BY SHA23JUL09/1448 OR CXL CA1501 G02AUG
 9.SSR TKNE CA HK1 PEKSHA 1501 G02AUG 99996****1194/1/P1
10.OSI CA CTCT 158012828                                --①
```

➢ IBE 实现提取特殊服务组信息

```
/*
 * 实现 PNR 旅客信息提取
 * */
/*生成 RT 对象*/
  RT   rtExample=new RT();

  /*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("RWGW2”);



 System.out.println("显示 OSI 信息:");
for(int i=0;i<rtres.getOSIsCount();i++){
        PNROSI pnrOsi=(PNROSI)rtres.getOSIAt(i);
         System.out.print("PNR 序号:"+pnrOsi.getIndex());
         System.out.print("     OSI 信息"+pnrOsi.getOsi());
         System.out.println("    需要 OSI 旅客编号:"+pnrOsi.getPNum());
    }
```

PNR OSI 信息查询显示结果

```
显示 OSI 信息:
PNR 序号:10     OSI 信息 CA CTCT 15801282879/P1     需要 OSI 旅客编号:P1
```

4.9. 提取运价信息

eTerm 中的运价信息：

```
RT:RWGW2

**ELECTRONIC TICKET PNR**
 1.韩杨 RWGW2
 2.  CA1501 G   SU02AUG  PEKSHA RR1   0830 1040          E
 3.SHA/T SHA/T 021-61327000/JEBSEN AIR SERVICE CO.,LTD/ZHOU ZHI YANG ABCDEFG
 4.ZHY FOR NOVOZYMES-BJ-T
 5.T
 6.SSR FOID CA HK1 NI****************/P1
 7.SSR OTHS CA 3CN315
 8.SSR ADTK 1E BY SHA23JUL09/1448 OR CXL CA1501 G02AUG
 9.SSR TKNE CA HK1 PEKSHA 1501 G02AUG 99996****1194/1/P1
10.OSI CA CTCT 158012828

pn1
11.RMK MP 15801282879/P1                                                 -
12.RMK BUKUSN 1116635/P1
13.RMK CA/EMS5T
14.RMK AUTOMATIC FARE QUOTE
15.FN/A/FCNY680.00/SCNY680.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY730.
00                                                                    --①
16.TN/999-96****1194/P1
17.FP/CASH,CNY/*3CN315                                                --②
18.SHA178
```

- IBE 实现提取运价信息

```
/*
 * 实现 PNR 旅客信息提取
 * */
/*生成 RT 对象*/
   RT   rtExample=new RT();

   /*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("SK49Z”);



 /*显示运价信息*
System.out.println("显示运价信息 FN:");
for(int i=0;i<rtres.getFNsCount();i++){
    PNRFN pnrFn=(PNRFN)rtres.getFNAt(i);
   System.out.println("PNR 序号:"+pnrFn.getIndex());
   System.out.println("     FN 信息"+pnrFn.getFn());
   System.out.print("     FCNY"+pnrFn.getAmount(BookFN.F));
   System.out.println("     货币单位"+pnrFn.getCurrency(BookFN.F));
   System.out.print("     ACNY"+pnrFn.getAmount(BookFN.A));
   System.out.println("     货币单位"+pnrFn.getCurrency(BookFN.A));
      System.out.print("     SCNY"+pnrFn.getAmount(BookFN.S));
      System.out.println("     货币单位"+pnrFn.getCurrency(BookFN.S));
      System.out.print("     XCNY"+pnrFn.getAmount(BookFN.X));
      System.out.println("     货币单位"+pnrFn.getCurrency(BookFN.X));
      for(int j=0;j<pnrFn.getTaxCnt();j++){
         System.out.print("第"+(j+1)+"项税"+pnrFn.getTaxAmount(BookFN.T, j));
         System.out.println("    货币单位"+pnrFn.getTaxCurrency(BookFN.T,j));
         }
      }
      System.out.println();
      System.out.println("显示运价信息 FP:");
       for(int i=0;i<rtres.getFpsCount();i++){
       PNRFP pnrFP=(PNRFP)rtres.getFpAt(i);
       System.out.println("PNR 序号:"+pnrFP.getIndex());
       System.out.println("     FP 代码:"+pnrFP.getPaytype());
       System.out.println("     货币单位:"+pnrFP.getCurrency());
       System.out.println("     FP 注释:"+pnrFP.getRemark());
   }
```

PNR 运价信息查询显示结果

```
显示运价信息 FN:
PNR 序号:15
    FN 信息 FN/A/FCNY680.00/SCNY680.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY730.00
    FCNY680.0    货币单位 CNY
    ACNY730.0    货币单位 CNY
    SCNY680.0    货币单位 CNY
    XCNY50.0    货币单位 CNY
第 1 项税 50.0    货币单位 CNY
第 2 项税-1.0    货币单位

显示运价信息 FP:
PNR 序号:17
    FP 代码:CASH
    货币单位:CNY
    FP 注释:FP/CASH,CNY/*3CN315
```

4.10. 提取 REMARK

eTerm 中的 REMARK 信息:

```
RT:RWGW2

**ELECTRONIC TICKET PNR**
 1.韩杨 RWGW2
 2.  CA1501 G   SU02AUG  PEKSHA RR1   0830 1040          E
 3.SHA/T SHA/T 021-61327000/JEBSEN AIR SERVICE CO.,LTD/ZHOU ZHI YANG ABCDEFG
 4.ZHY FOR NOVOZYMES-BJ-T
 5.T
 6.SSR FOID CA HK1 NI****************/P1
 7.SSR OTHS CA 3CN315
 8.SSR ADTK 1E BY SHA23JUL09/1448 OR CXL CA1501 G02AUG
 9.SSR TKNE CA HK1 PEKSHA 1501 G02AUG 99996****1194/1/P1
10.OSI CA CTCT 158012828

□pn1
11.RMK MP 15801282879/P1                                          --①
12.RMK BUKUSN 1116635/P1                                          --②
13.RMK CA/EMS5T                                                   --③
14.RMK AUTOMATIC FARE QUOTE                                       --④
15.FN/A/FCNY680.00/SCNY680.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY730.
00
16.TN/999-96****1194/P1
17.FP/CASH,CNY/*3CN315
18.SHA178
```

➢ IBE 实现提取 REMARK 信息

```
/*
 * 实现 PNRREMARK 信息提取
 * */
/*生成 RT 对象*/
   RT   rtExample=new RT();

   /*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("SK49Z”);



 /*显示 REMARK 信息组*/
 System.out.println("显示 REMARK 信息组:");
    for(int i=0;i<rtres.getRMKsCount();i++){
   PNRRMK pnrRmk=(PNRRMK)rtres.getRMKAt(i);
    System.out.print("PNR 序号:"+pnrRmk.getIndex());
    System.out.println("     REMARK 内容:"+pnrRmk.getRmkinfo());
 }
```

PNR REMRAK 组信息查询显示结果

```
显示 REMARK 信息组:
PNR 序号:11     REMARK 内容:MP 15801282879/P1
PNR 序号:12     REMARK 内容:BUKUSN 1116635/P1
PNR 序号:13     REMARK 内容:CA/EMS5T
PNR 序号:14     REMARK 内容:AUTOMATIC FARE QUOTE
```

4.11. 提取票号组

eTerm 中的票号组信息：

```
RT:PBK5Y

  **ELECTRONIC TICKET PNR**
 1.王铮亮 2.赵郴 PBK5Y
 3.  CA1501 G   SU02AUG  PEKSHA RR2   0830 1040          E T3--
 4.BJS/T BJS/T 010-65071234/PEK HONG YUN DA AVATION SERVICE CO.,LTD/DAI JIAN
    DONG78735-44 ABCDEFG
 5.81918829 BJS-S0903119
 6.T
 7.SSR FOID CA HK1 NI******************/P2
 8.SSR FOID CA HK1 NI******************/P1
 9.SSR ADTK 1E BY BJS21JUL09/2017 OR CXL CA1501 G02AUG
10.SSR TKNE CA HK1 PEKSHA 1501 G02AUG 99996****9932/1/P2
11.SSR TKNE CA HK1 PEKSHA 1501 G02AUG 99996****9931/1/P1
12.RMK BLK PNR                                                  -
13.RMK CA/B9W3J
14.RMK AUTOMATIC FARE QUOTE
15.FN/A/FCNY680.00/SCNY680.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY730.
00
16.TN/999-96****9931/P1                                   --①
17.TN/999-96****9932/P2                                   --②
18.FP/CASH,CNY
19.BJS584
```

- IBE 实现提取票号组信息

```
/*
 * 实现 PNR 旅客信息提取
 * */
/*生成 RT 对象*/
   RT   rtExample=new RT();

   /*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("PB");


     System.out.println("显示出票组信息:");
     for(int i=0;i<rtres.getTktnosCount();i++){
          PNRTktNo pnrTktNo=(PNRTktNo)rtres.getTktnoAt(i);
            System.out.print("PNR 序号:"+pnrTktNo.getIndex());
            System.out.print("     票号:"+pnrTktNo.getTktNo());
            System.out.println("票号对应的旅客序号:"+pnrTktNo.getPsgrID());
     }
```

PNR 票号组信息查询显示结果

显示票号组信息:
*PNR 序号:16　　票号:999-96****9931　　票号对应的旅客序号:P1*
*PNR 序号:17　　票号:999-96****9932　　票号对应的旅客序号:P2*

4.12. 提取责任组

eTerm 中的责任组信息:

```
RT:RWGW2

**ELECTRONIC TICKET PNR**
 1.韩杨 RWGW2
 2.  CA1501 G   SU02AUG  PEKSHA RR1   0830 1040          E
 3.SHA/T SHA/T 021-61327000/JEBSEN AIR SERVICE CO.,LTD/ZHOU ZHI YANG ABCDEFG
 4.ZHY FOR NOVOZYMES-BJ-T
 5.T
 6.SSR FOID CA HK1 NI****************/P1
 7.SSR OTHS CA 3CN315
 8.SSR ADTK 1E BY SHA23JUL09/1448 OR CXL CA1501 G02AUG
 9.SSR TKNE CA HK1 PEKSHA 1501 G02AUG 99996****1194/1/P1
10.OSI CA CTCT 158012828

pn1
11.RMK MP 15801282879/P1                                         -
12.RMK BUKUSN 1116635/P1
13.RMK CA/EMS5T
14.RMK AUTOMATIC FARE QUOTE
15.FN/A/FCNY680.00/SCNY680.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY730.
00
16.TN/999-96****1194/P1
17.FP/CASH,CNY/*3CN315
18.SHA178                                                      --①
```

➢ IBE 实现提取 REMARK 信息

```
/*
 *  实现 PNRREMARK 信息提取
 * */
/*生成 RT 对象*/
   RT   rtExample=new RT();

   /*提取 PNR 信息*/
RTResult rtres = rtExample. retrieve ("SK49Z”);


 /*显示责任组信息组*/
 System.out.println("显示责任组信息");
 PNRResp pnrResp=rtres.getResp();
 System.out.println("PNR 序号:"+pnrResp.getIndex());
 System.out.println("PNR  责任组:"+pnrResp.getOfficecode());
```

PNR 责任组信息查询显示结果

```
显示责任组信息
PNR 序号:18
PNR  责任组:SHA178
```

5. PNR 修改

实现类：com.travelsky.ibe.client.pnr.PnrManage

5.1. 删除订座记录

➢ eTerm 删除订座记录过程

eTerm 中运用主机指令删除本 OFFICE 订座记录如下：

```
>RT ZY47S  --①
 1.IBETESTER/IBEGROUP ZY47S/MU
 2.  CA1501 Y  WE26MAY  PEKSHA HK1   0830 1040     E T3--
 3.010-84669396
 4.TL/1200/25MAY/PEK099
 5.SSR FOID CA HK1 NI123456789/P1
 6.SSR ADTK CA BY PEK23MAY10/0830 OR CXL CA1501 Y26MAY
 7.PEK099
>XEPNR@                    --②
PNR CANCELLED ZY47S
```

eTerm 中运用主机指令删除非本 OFFICE 的订座记录如下：
当 BJS191 的终端配置需要删除 BJS280 预订的订座记录时

```
>RT  XD40F
 1.IBETESTER/IBEGROUP XD40F
 2.  CA984  Y   FR26JUN  PEKPVG HK1   0720 0930       E T3--
 3.BJS/T BJS/T 010-64489371/FEAS XI BA HE BOOKING OFFICE/WU YAN ABCDEFG
 4.NC
 5.TL/1200/24JUN/BJS280
 6.SSR ADTK 1E BY BJS11JUN09/1038 OR CXL CA 984 Y26JUN
 7.RMK CA/ZY481
>XEPNR@
非本 OFFICE 的 PNR,请检查 PNR 号码!如确要取消,请输入责任组 OFC
>XEPNR@BJS280
PNR CANCELLED XD40F
```

同时系统会自动向 BJS280 的 RP 或者 GQ 信箱中发一封信，通知订座记录被删除。

```
PNR XD40F WAS CANCELLED BY AGENT8888  PID7613  BJS191 1040 28MAY09
```

➢ IBE 实现删除订座记录过程

IBE 在处理时是通过 deletePnr 接口实现了删除 PNR 的操作，在后台相当于把提取订座记录和后续的删除操作打包在一块，通过一个原子性事务来实现了删除 PNR 的操作。
com.travelsky.ibe.client.pnr.PnrManager 类:
deletePnr(String pnrno)删除本 OFFICE 的订座记录的方法
deletePnr(String pnrno,String office)删除其他 OFFICE 的订座记录的方法

需要注意的是 IBE 的权限要求,用户只能够删除本 OFFICE 的预订的订座记录。如需要支持删除其他 OFFICEPNR 需要提出额外的申请。
删除 PNR 前

```
>RTN/TDW08
0.10HELLO NM2 TDW08/ZH
1.成人甲 2.成人乙
3.  ZH9890 Y   SA01AUG  PEKSZX HN10  0800 1110      E T2B
4.NC
5.SZH/T
6.TL/0000/31JUL/SZH314
7.SZH314
```

删除 PNR 后

```
 >RTN/TDW08
   *THIS PNR WAS ENTIRELY CANCELLED*
003     SZH314 4960 0814 20JUL IK
001 X0.10HELLO NM2 TDW08/ZH
    X1.成人甲(002) X2.成人乙(002)
001 X3.  ZH9890 Y   SA01AUG  PEKSZX XX10  0800 1110      E T2B
      NN(001)  HN(001)  XX(003)
001 X4.BKD 10 CNL 2 SPLIT 0
001 X5.NC
001 X6.SZH/T
001 X7.TL/0000/31JUL/SZH314
001  8.SZH314
```

```
String pnrNo = "XD441";
PnrManage manager = new PnrManage();
String res = manager. deletePnr(pnr);
```

5.2. 删除指定编号的组

IBE 在处理时 PNR 是通过 deleteItem 接口实现了删除 PNR 的指定编号的组，相当于把提取订座记录和后续的删除操作打包在一块,通过一个原子性事务来实现了删除指定编号的组的操作。

deleteItem(String pnrno, int[] indexes)
需要注意的是删除组后，一定要保证 PNR 有最基本的元素，航线组，联系组，出票时限至少要保留一个，旅客姓名组不在本接口操作范围之内。

删除 PNR 前

```
>RTN/ED96X
 1.陈菲 2.王杰 ED96X
 3.  ZH9890 H   TU21JUL  PEKSZX HK2   0800 1110     E T2B
 4.T SZX/SZX/T-0755-27858888/SZX SHEN ZHOU AVIATION KE YUN AGE
 5.T SZX/NCY. /JIANG XING QI
 6.SSR TKTL ZH SS/ SZX 1800/20JUL
 7.OSI CA CTC 13510221161 WLF
 8.PEK1E/TLXML/SZX747
```

终端操作

```
>RTN/ED96X
 1.陈菲 2.王杰 ED96X
 3.  ZH9890 H   TU21JUL  PEKSZX HK2   0800 1110     E T2B
 4.T SZX/SZX/T-0755-27858888/SZX SHEN ZHOU AVIATION KE YUN AGE
 5.T SZX/NCY. /JIANG XING QI
 6.SSR TKTL ZH SS/ SZX 1800/20JUL
 7.OSI CA CTC 13510221161 WLF
 8.PEK1E/TLXML/SZX747
>XE:7
@
  ZH9890  H TU21JUL  PEKSZX HK2   0800 1110   6
ED96X
```

删除 PNR 的第 7 项后

```
>RTN/ED96X
 1.陈菲 2.王杰 ED96X
 3.  ZH9890 H   TU21JUL  PEKSZX HK2   0800 1110     E T2B
 4.T SZX/SZX/T-0755-27858888/SZX SHEN ZHOU AVIATION KE YUN AGE
 5.T SZX/NCY. /JIANG XING QI
 6.SSR TKTL ZH SS/ SZX 1800/20JUL
 7.PEK1E/TLXML/SZX747
```

IBE 指令调用

```
String pnrNo = "ED96X";
PnrManage manager = new PnrManage();
String res = manager. deleteItem(pnr, new int[7]);
```

5.3. 为 PNR 添加新组

IBE 在处理时 PNR 是通过 addPnrInfo 接口实现了在已有的 PNR 中添加新组。

addPnrInfo(String pnrno, BookInfomation bookinfo)
需要注意的是对于非团队票，姓名组不能添加。

在 BookInfomation 可以添加和 SellSeat 中定义的各种组，关于组的说明可以参照 SellSeat 的功能。

IBE 指令调用

```
String pnrNo = "ED96X";
BookInfomation bi = new BookInfomation();
bi.addContact("Hello world");
PnrManage manager = new PnrManage();
String res = manager. addPnrInfo(pnr, new int[7]);
```

5.4. 航班改期

在预订了订座记录之后，可能需要座位也可能发生变化。这是只能在一次封口中完成，IBE 提供一套指令来完成航班改期。

String changeAirSeg(String pnr, BookAirSeg oldSeg, BookAirSeg newSeg)

String changeAirSeg(String pnr, Vector oldSegs, Vector newSegs)

改期前

```
>RTN/ ZY40C
  1.TRAVELSKY/TEST ZY40C/ZH
  2.  ZH9890 Y   SA01AUG  PEKSZX HK1   0800 1110     E T2B
  3.NC
  4.SZH/T
  5.TL/0000/31JUL/SZH314
  6.SZH314
```

改期的终端操作

```
>RTN/ZY40C
  1.TRAVELSKY/TEST ZY40C/ZH
  2.  ZH9890 Y   MO10AUG  PEKSZX HK1   0800 1110     E T2B
  3.NC
  4.SZH/T
  5.TL/0000/31JUL/SZH314
  6.SZH314
>xE:2
SS:ZH9890/Y/10AUG09/PEKSZX/NN1
@ik
>RTN/ZY40C
  1.TRAVELSKY/TEST ZY40C/ZH
  2.  ZH9890 Y   MO10AUG  PEKSZX HK1   0800 1110     E T2B
  3.NC
  4.SZH/T
  5.TL/0000/31JUL/SZH314
  6.SZH314
```

IBE 指令调用

```
SellSeat ss = new SellSeat();
ss.addPassenger(new BookPassenger("travelsky/test"));
ss.addAirSeg("ZH9890",'Y', "PEK", "SZX", "NN", 1, QDateTime.stringToDate("20090801", "yyyymmdd"));
ss.addContact("NC");
ss.setTimelimit(QDateTime.stringToDate("20090731", "yyyymmdd"));
String pnr = ss.commit();
//改期操作
PnrManage pnrman = new PnrManage();
BookAirSeg oldseg = new BookAirSeg("ZH9890",'Y', "PEK", "SZX", "NN", 1, QDateTime.stringToDate("20090801", "yyyymmdd"));
BookAirSeg newseg = new BookAirSeg("ZH9890",'Y', "PEK", "SZX", "NN", 1, QDateTime.stringToDate("20090810", "yyyymmdd"));
pnrman.changeAirSeg(pnr, oldseg, newseg);
```

5.5. 修改出票组

出票组不能通过常规的删除/添加实现。IBE 提供了 changeTkt(String, BookInfomation)支持修改 PNR 的出票组。

需要添加新的出票组填加在 bookinfo 参数中。此参数 bookinfo 中唯一有效的是出票状态组,其他项目均无效。使用 BookInfomation.addTktstatus(BookTktStatus)方法添加可以添加多个，每个对应 pnr 中一个旅客，也可只添加一个对应 pnr 中所有旅客的 出票状态组。 会主动删除旧有对应旅客的出票组。如果添加的出票状态组中指定适用该组的旅客,则需要确认写入的姓名与 pnr 返回：正常结果"OK"。

修改前

```
>RT: ZYD0F
 1.TRAVELSKY/TEST ZYD0F/ZH
 2.  ZH9890 Y   SA01AUG  PEKSZX HK1   0800 1110     E T2B
 3.NC
 4.SZH/T
 5.TL/0000/31JUL/SZH314
 6.SZH314
```

修改的终端操作

```
>RT: ZYD0F
 1.TRAVELSKY/TEST ZYD0F/ZH
 2.  ZH9890 Y   SA01AUG  PEKSZX HK1   0800 1110     E T2B
 3.NC
 4.SZH/T
 5.TL/0000/31JUL/SZH314
 6.SZH314
```

```
>XE/5
TK:T1/0000/30JUL09/SZH314
@i
>RT: ZYD0F
1.TRAVELSKY/TEST ZYD0F/ZH
 2.  ZH9890 Y   SA01AUG  PEKSZX HK1   0800 1110      E T2B
 3.NC
 4.SZH/T
 5.TL/0000/30JUL/SZH314
 6.SZH314
```

IBE 指令调用

```
SellSeat ss = new SellSeat();
ss.addPassenger(new BookPassenger("travelsky/test"));
ss.addAirSeg("ZH9890",'Y',    "PEK",    "SZX",    "NN",    1,    QDateTime.stringToDate("20090801", "yyyymmdd"));
ss.addContact("NC");
ss.setTimelimit(QDateTime.stringToDate("20090731", "yyyymmdd"));
String pnr = ss.commit();
//修改出票时限操作
PnrManage pnrman = new PnrManage();
BookInfomation bi = new BookInfomation();
bi.addTktstatus(new BookTktStatus(QDateTime.stringToDate("20090730", "yyyymmdd")));
pnrman.changeTkt(pnr, bi);
```

5.6. 分离 PNR

多人 PNR 或者团队票 PNR 可以将部分旅客和空白座位拆分成一个新的 PNR。

IBE 提供了 splitPNR(String pnrno, java.util.Vector passengers, int count)支持分离 PNR。

执行操作前

```
>RTN/TE405
 0.10HELLO NM3 TE405/ZH
  1.成人丙 2.成人甲 3.成人乙
  4.  ZH9890 Y   SA01AUG  PEKSZX HN10  0800 1110      E T2B
  5.NC
  6.SZH/T
  7.TL/0000/31JUL/SZH314
 8.SZH314
```

执行操作后，10 个人的团变成 5 个人团

```
> RTN/TDW0W
0.5HELLO NM1 TE40W/ZH
 1.成人丙
 2.  ZH9890 Y   SA01AUG  PEKSZX HN5   0800 1110      E T2B
 3.NC
 4.SZH/T
 5.TL/0000/31JUL/SZH314
 6.SZH314
```

同时还新生成一个 PNR

```
> RTN/TDW0E
0.5HELLO NM2 TDW0E
 1.成人甲 2.成人乙
 3.  ZH9890 Y   SA01AUG  PEKSZX HN5   0800 1110      E T2B
 4.NC
 5.SZH/T
 6.TL/0000/31JUL/SZH314
 7.SZH314
```

```
//创建空的团队票 PNR
SellSeat ss = new SellSeat();
ss.setGroupName("Hello");
ss.setPassengerNumber(10);
ss.setGroupTicket(true);
ss.addAirSeg("ZH9890",'Y', "PEK", "SZX", "NN", 10, QDateTime.stringToDate("20090801", "yyyymmdd"));
ss.addContact("NC");
ss.setTimelimit(QDateTime.stringToDate("20090731", "yyyymmdd"));
String pnr = ss.commit();
System.out.println(pnr);
//添加旅客姓名
Vector psgrs = new Vector();
BookPassenger psgr = new BookPassenger("成人甲");
BookPassenger psgr2 = new BookPassenger("成人乙");
BookPassenger psgr3 = new BookPassenger("成人丙");
psgrs.add(psgr);
psgrs.add(psgr2);
psgrs.add(psgr3);
PnrManage pnrman = new PnrManage();
String res = pnrman.addGroupName(pnr, psgrs);
System.out.println(pnr);
//分离 PNR
Vector psgrs2 = new Vector();
```

```
BookPassenger psgr = new BookPassenger("成人甲");
BookPassenger psgr2 = new BookPassenger("成人乙");
psgrs2.add(psgr);
psgrs2.add(psgr2);
PnrManage pnrman = new PnrManage();
String res = pnrman.splitPNR(pnr, psgrs2, 3);
System.out.println(res);//新 PNR
```

5.7. 座位再确认

在预订了订座记录之后，座位也可能发生变化，这些变化就反映在航段的行动代码上。通过查看行动代码，可以确定座位的真实占座情况。

如行动代码为 NN，表示要申请座位，系统在接受申请后可能返回 HK，就是表示接受这个申请，在该航班上预留了座位给订座人，也可能返回 HL，就是现在没有座位，不过有合适的座位时，系统会自动占上这个座位。

在日常使用中比较重要的行动代码：

NN，表示提出申请需要占座

HK，KL，KK，TK，DK，表示占座成功。

RR，表示已经出票或者即将出票

在成功出票以后，需要对座位进行再确认。

- eTerm 进行座位再确定过程

eTerm 中运用主机指令对已经出票的 PNR 进行再确认过程如下：

```
> RT XD441
 1.IBETESTER/IBEGROUP XD441
 2.  CA1501 Y   WE26MAY  PEKSHA HK1   0830 1040         E T3--        ①
 3.010-84669396
 4.T
 5.SSR FOID CA HK1 NI123456789/P1
 6.SSR ADTK 1E BY PEK23MAY10/0830 OR CXL CA1501 Y26MAY
 7.RMK CA/ZY4KS
 8.PEK099
> 2/RR           ②
@ik
  CA1501  Y FR26JUN  PEKSHA RR1   0830 1040
XD441 -  □航空公司使用自动出票时限，请检查 PNR□
> RT XD441
1.IBETESTER/IBEGROUP XD441
 2.  CA1501 Y   WE26MAY  PEKSHA RR1   0830 1040         E T3--
 3.010-84669396
 4.T
 5.SSR FOID CA HK1 NI123456789/P1
 6.SSR ADTK 1E BY PEK23MAY10/0830 OR CXL CA1501 Y26MAY
 7.RMK CA/ZY4KS
```

```
8.PEK099
```

- IBE 实现座位再确定过程

IBE 在处理时是通过 reconfirmAirSeg 接口实现了座位再确认的操作，在后台相当于把提取订座记录和后续的再确认操作打包在一块，通过一个原子性事务来实现了再确认 PNR 的操作。

com.travelsky.ibe.client.pnr.PnrManager 类：

String reconfirmAirSeg(String pnrno, PNRAirSeg segment, int force)

pnrno 是需要做再确认的订座记录编号

segment 是需要再确认的航段，如果是全 pnr 所有航段都需要 rr,则使用 null 作为参数。否则，在选择航段时检查其中的航班号，日期

对于选项 force 的解释如下：

RECONFIRM_ALL_ACTION:

尝试将要求 RR 的所有航段置 RR，不管原来的 ActionCode 是什么

RECONFIRM_ALL_POSSIBLE_ACTION:

尝试将要求 RR 的航段中所有可能 RR 的航段置 RR，即原来的 ActionCode 是 KK,HK,TK,KL 的航段，但即使要求 RR 的航段状态不是 KK,KL,TK,HK,RR 中的一种也不报错

RECONFIRM_CHECK_STRICTLY:

尝试将要求 RR 的航段置 RR，即原来的 ActionCode 是 KK,HK,TK,KL 的航段，如果要求 RR 的航段中有不是 KK,HK,TK,KL,RR 状态的航段,将抛出异常

对所有航段的可能状态进行再确认

```
String pnrNo = "XD441";
PnrManage manager = new PnrManage();
String res = manager.reconfirmAirSeg(pnrNo,
                                      null,
                                     PnrManage.RECONFIRM_ALL_POSSIBLE_ACTION);
if("OK".equals(res)){
        //...座位再确认成功
}else if(res.startsWith("Error:")){
        //...座位再确认失败，请重新提取订座记录
}
```

对指定的航段进行再确认

```
RT rt = new RT();
String pnrNo = "XD441";
/*先提取 PNR 的详细信息*/
```

```
        RTResult pnrResult = rt.retrieve(pnrNo);
    /*获得再确认需要的航段*/
        PNRAirSeg segment = pnrResult.getAirSegAt(0);
        if(!"RR".equals(segment.getActionCode())){
            PnrManage manager = new PnrManage();
            String res = manager.reconfirmAirSeg(pnrNo,
                                segment,
                                PnrManage.RECONFIRM_ALL_POSSIBLE_ACTION);
            if("OK".equals(res)){
                //...座位再确认成功
            }else if(res.startsWith("Error:")){
                //...座位再确认失败，请重新提取订座记录
            }
        }
```

5.8. 团队票添加旅客姓名

团队票可以在生成 PNR 之后，再输入具体的旅客姓名信息。

IBE 提供了 addGroupName(String pnrno, java.util.Vector passengers) 支持向团队票中加入旅客姓名。

passengers 为需要添加的旅客信息，该 Vector 对象中每个元素均为一个 BookPassenger 对象实例。

执行操作前

```
> RTN/TDW08
0.10HELLO NM0 TDW08/ZH
 1.  ZH9890 Y   SA01AUG  PEKSZX HN10  0800 1110      E T2B
 2.NC
 3.SZH/T
 4.TL/0000/31JUL/SZH314
 5.SZH314
```

执行操作后

```
 >RTN/TDW08
 0.10HELLO NM2 TDW08/ZH
 1.成人甲 2.成人乙
 3.  ZH9890 Y   SA01AUG  PEKSZX HN10  0800 1110      E T2B
 4.NC
 5.SZH/T
 6.TL/0000/31JUL/SZH314
 7.SZH314
```

```
        //创建空的团队票PNR
```

```
            SellSeat ss = new SellSeat();
            ss.setGroupName("Hello");
            ss.setPassengerNumber(10);
            ss.setGroupTicket(true);
            ss.addAirSeg("ZH9890",'Y',   "PEK",   "SZX",   "NN",   10,   QDateTime.stringToDate("20090801",
"yyyymmdd"));
            ss.addContact("NC");
            ss.setTimelimit(QDateTime.stringToDate("20090731", "yyyymmdd"));
            String pnr = ss.commit();
            System.out.println(pnr);
            //添加旅客姓名
            Vector psgrs = new Vector();
            BookPassenger psgr = new BookPassenger("成人甲");
            BookPassenger psgr2 = new BookPassenger("成人乙");
            psgrs.add(psgr);
            psgrs.add(psgr2);
            PnrManage pnrman = new PnrManage();
            String res = pnrman.addGroupName(pnr, psgrs);
            System.out.println(pnr);
```

5.9. 团队票删除旅客姓名

团队票可以对不能成行的旅客删除，同时释放座位。

IBE 提供了 removeName(String pnrno, java.util.Vector passengers) 支持向团队票中删除旅客姓名。

passengers 为需要添加的旅客信息，该 Vector 对象中每个元素均为一个 BookPassenger 对象实例。

执行操作前

```
>RTN/TDW0Q
 0.10HELLO NM2 TDW0Q/ZH
 1.成人甲 2.成人乙
 3.  ZH9890 Y   SA01AUG  PEKSZX HN10  0800 1110     E T2B
 4.NC
 5.SZH/T
 6.TL/0000/31JUL/SZH314
 7.SZH314
```

执行操作后，10 个人的团变成 8 个人

```
> RTN/TDW0Q
0.8HELLO NM0 TDW0Q/ZH
 1.  ZH9890 Y   SA01AUG  PEKSZX HN10  0800 1110     E T2B
 2.NC
 3.SZH/T
 4.TL/0000/31JUL/SZH314
 5.SZH314
```

```
        //创建空的团队票PNR
        SellSeat ss = new SellSeat();
        ss.setGroupName("Hello");
        ss.setPassengerNumber(10);
        ss.setGroupTicket(true);
        ss.addAirSeg("ZH9890",'Y',   "PEK",   "SZX",   "NN",   10,   QDateTime.stringToDate("20090801",
"yyyymmdd"));
        ss.addContact("NC");
        ss.setTimelimit(QDateTime.stringToDate("20090731", "yyyymmdd"));
        String pnr = ss.commit();
        System.out.println(pnr);
        //添加旅客姓名
        Vector psgrs = new Vector();
        BookPassenger psgr = new BookPassenger("成人甲");
        BookPassenger psgr2 = new BookPassenger("成人乙");
        psgrs.add(psgr);
        psgrs.add(psgr2);
        PnrManage pnrman = new PnrManage();
        String res = pnrman.addGroupName(pnr, psgrs);
        System.out.println(pnr);
        //删除旅客
        res = pnrman.removeName(pnr, psgrs);
        System.out.println(pnr);
```

5.10. 团队票减少座位

团队票可以未输入旅客姓名的座位进行释放。

IBE 提供了 reduceGroupSeats(String pnrno, int count)支持向团队票中减少座位。

执行操作前

```
>RTN/TDW0W
 0.10HELLO NM2 TDW0W/ZH
 1.成人甲 2.成人乙
 3.  ZH9890 Y   SA01AUG  PEKSZX HN10  0800 1110     E T2B
 4.NC
 5.SZH/T
 6.TL/0000/31JUL/SZH314
 7.SZH314
```

执行操作后，10 个人的团变成 7 个人团

```
 > RTN/TDW0W
0.7HELLO NM2 TDW0W/ZH
 1.成人甲 2.成人乙
 3.  ZH9890 Y   SA01AUG  PEKSZX HN7   0800 1110     E T2B
 4.NC
 5.SZH/T
 6.TL/0000/31JUL/SZH314
 7.SZH314
```

```
//创建空的团队票 PNR
SellSeat ss = new SellSeat();
ss.setGroupName("Hello");
ss.setPassengerNumber(10);
ss.setGroupTicket(true);
ss.addAirSeg("ZH9890",'Y',   "PEK",   "SZX",   "NN",   10,   QDateTime.stringToDate("20090801", "yyyymmdd"));
ss.addContact("NC");
ss.setTimelimit(QDateTime.stringToDate("20090731", "yyyymmdd"));
String pnr = ss.commit();
System.out.println(pnr);
//添加旅客姓名
Vector psgrs = new Vector();
BookPassenger psgr = new BookPassenger("成人甲");
BookPassenger psgr2 = new BookPassenger("成人乙");
psgrs.add(psgr);
psgrs.add(psgr2);
PnrManage pnrman = new PnrManage();
String res = pnrman.addGroupName(pnr, psgrs);
System.out.println(pnr);
//删除无用的座位
res = pnrman. reduceGroupSeats(pnr, 3);
```

6. PATA 运价查询

6.1. 查询 PATA 运价

- eTerm PATA 查询过程

  1.在主机中预定一个 PNR,然后进行 RT：

```
RT: W7H06
```

```
RT W7H06
 1.TEST/TEST W7H06
 2.  CA1831 Y   TU30MAR  PEKSHA HK1   0730 0940         E T3T2
 3.123456
 4.TL/0000/29MAR/PEK099
 5.SSR FOID CA HK1 NI123456789/P1
 6.SSR ADTK 1E BY PEK27MAR10/0730 OR CXL CA1831 Y30MAR
 7.RMK CA/L400E
 8.PEK099
```

  2. 执行 PAT:A 指令，得到运价：

```
PAT: A
```

```
>PAT:A
01 PART2TESTTCER FARE:CNY230.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:680.00
SFC:01
02 Y FARE:CNY300.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:750.00
SFC:02
03 YB90 FARE:CNY810.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1260.00
SFC:03
04 YB90 FARE:CNY900.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1350.00
SFC:04
05 YB90 FARE:CNY900.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1350.00
SFC:05
06 YB FARE:CNY1000.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1450.00
SFC:06
07 YB FARE:CNY1000.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1450.00
SFC:07
08 FRB/Y FARE:CNY1080.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1530.00
```

```
□SFC:08
```

3．如果需要将某条运价录入到 PNR，则实行 SFC: 序号

```
SFC:01
```

```
 1.TEST/TEST W7H06
 2.  CA1831 Y   TU30MAR  PEKSHA HK1   0730 0940         E T3T2
 3.123456
 4.TL/0000/29MAR/PEK099
 5.FC/A/PEK CA SHA 230.00PART2TESTTCER CNY230.00END
 6.SSR FOID CA HK1 NI123456789/P1
 7.SSR ADTK 1E BY PEK27MAR10/0730 OR CXL CA1831 Y30MAR
 8.RMK CA/L400E
 9.RMK AUTOMATIC FARE QUOTE
10.FN/A/FCNY230.00/SCNY230.00/C3.00/XCNY450.00/TCNY50.00CN/TCNY400.00YQ/
   ACNY680.00
11.TC/A/ADELETC2345678                                                 +
```

可见运价 FN， FC 项增加了 A 标识

- IBE 的 PATA 接口使用过程

实现类： com.travelsky.ibe.client.pnr.PAT
com.travelsky.ibe.client.pnr.PATFareItem
com.travelsky.ibe.client.pnr.PATResult

对于 PATA 的查询 IBE 提供了两种方法，分别是：
通过 PNR 查询，并且可以录入
通过航段参数进行查询

通过 PNR 查询：

1. 调用方法：

```
public PATResult doPat(String pnrno, String option, String type,int index,
        boolean ifinput,String payment,String clientcode)
```

其中各参数的含义：

pnr	PNR 编号
option	如果查净价则写 A，如果查询南航产品舱位价则写产品名称，不写则填 null 或""
type	查询类型，IN：婴儿；CH：儿童；JC：因公带伤警察；GM：伤残军人,不写则填 null 或""
index	旅客序号,从 1 开始,查询净价时此项才有效
ifinput	是否将查询的运价结果录入 PNR，如果查询净价且为多个结果并要选定一个录入到 PNR，要以选定的结果集为参数再执行

doPatASecond 方法
payment　支付方式，信用卡:CC，现金:null 或 CASH
clientcode　大客户编码(6 位)，不指定则输入 null（不能为空字符串）

doPatASecond 方法：

```
public PATResult doPatASecond(PATResult patres_pre, String pnrno, int
    index,int psgid ,String type,String payment,String clientcode)
```

各参数的含义：（因为 doPatASecond 之前要调用 doPat 方法，所以两个方法的查询类参数要保持值一致）

patres_pre　为上次做 PATA 返回的 PATResult
pnr　PNR 编号
index　选定的某一条在 patres_pre 中净价信息 PATFareItem 的序号，从 0 开始
psgid　旅客序号,从 1 开始
type　查询类型，IN：婴儿；CH：儿童；JC：因公带伤警察；GM：伤残军人,不写则填 null 或""
payment　支付方式，信用卡:CC，现金:null 或 CASH
clientcode　大客户编码(6 位)，不指定则输入 null（不能为空字符串）

2. 查询结果与主机的对应

```
>PAT:A
01 PART2TESTTCER FARE:CNY230.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:680.00   ---①
□SFC:01
02 Y FARE:CNY300.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:750.00
□SFC:02
03 YB90 FARE:CNY810.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1260.00
□SFC:03
04 YB90 FARE:CNY900.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1350.00
□SFC:04
05 YB90 FARE:CNY900.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1350.00
□SFC:05
06 YB FARE:CNY1000.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1450.00
□SFC:06
07 YB FARE:CNY1000.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1450.00
□SFC:07
08 FRB/Y FARE:CNY1080.00 TAX:CNY50.00 YQ:CNY400.00  TOTAL:1530.00
□SFC:08
```

PATA 查询结果存放于 PATAResult 中，其中各项对应如下：

① 运价信息：　主机运价信息存放在 PATFareItem 中，通过 PATResult 中

的 PATFareItem getFareItem(int index)取得

PATFareItem 说明：

获得某条运价的序号：int getIndex()

获得某条运价文本： String getFare()

运价条数： PATFareItem 的条数通过 PATResult 中的 public int getFareNumber()获得

3. PATA 的使用用例

```
/*
*查询 PNR 的 PATA 运价，先不录入 PNR，查运价后再通过 dopatasencond()进行录入
* */

        PAT pat = new PAT();
        try{
            PATResult   res = pat.doPat("ZQ803","A",null,1,false,null,null);
            System.out.println(res);
        }catch(Exception e){
            e.printStackTrace();
        }
```

IBE 生成的 PATA 查询结果

```
0:Y FARE:CNY1130.00 TAX:CNY50.00 YQ:CNY400.00    TOTAL:1580.00
```

通过 dopataSecond 方法将运价录入到 PNR

```
/*
 *  查询 PNR 的 PATA 运价
* */
        PAT pat = new PAT();
        try{
            PATResult   res = pat.doPat("ZQ803","A",null,1,false,null,null);
            res = pat.doPatASecond(res,"ZQ803",0,1);
        }catch(Exception e){
            e.printStackTrace();
        }
```

操作成功后，运价录入到 PNR，主机 PNR 的信息：

```
rt ZQ803
 1.TEST/TEST ZQ803/SC
 2.  CA1831 Y   TU30MAR  PEKSHA HK1   0730 0940     E T2T2
 3.123456
 4.TL/0000/29MAR/PEK099
 5.FC/A/PEK CA SHA 1130.00Y CNY1130.00END/P1
 6.SSR FOID CA HK1 NI123456789/P1
 7.SSR ADTK CA BY PEK27MAR10/0730 OR CXL CA1831 Y30MAR
 8.RMK AUTOMATIC FARE QUOTE/P1
 9.FN/A/FCNY1130.00/SCNY1130.00/C0.00/XCNY450.00/TCNY50.00CN/TCNY400.00YQ/
   ACNY1580.00/P1
10.FP/CASH,CNY/P1
11.PEK099
```

支付方式为信用卡的用例：

```
/*
*查询 PNR 的 PATA 运价，先不录入 PNR，查运价后再通过 dopatasencond()进行录入
* */

        PAT pat = new PAT();
        try{
            PATResult    result = pat.doPat("HTR277","A",null,0,false,"CC",null);
            System.out.println(res);
        }catch(Exception e){
            e.printStackTrace();
        }
```

IBE 生成的 PATA 查询结果

```
0:Y FARE:CNY1700.00 TAX:CNY50.00 YQ:CNY70.00    TOTAL:1820.00
```

通过 dopataSecond 方法将运价录入到 PNR

```
/*
 *  查询 PNR 的 PATA 运价
 * */
        PAT pat = new PAT();
        try{
          PATResult result1 =
         pat.doPatASecond(result,"HTR277",0,1,null,"CC",null);
         res = pat.doPatASecond(res,"ZQ803",0,1);
        }catch(Exception e){
            e.printStackTrace();
```

```
            }
```

操作成功后，运价录入到 PNR，主机 PNR 的信息：

```
1.勃艮第 HTR277
 2.  CA1351 Y   TH30DEC  PEKCAN HK1   0800 1100         E T3--
 3.BJS/T BJS/T-010-82861011/BEIJING LONG CHANG DA SCIENCE LTD.,CO/YIN YINGFENG
    ABCDEFG
 4.1234567
 5.TL/0000/29DEC/BJS734
 6.FC/A/PEK CA CAN 1700.00Y CNY1700.00END/P1
 7.SSR ADTK 1E BY BJS27DEC10/0800 OR CXL CA1351 Y30DEC
 8.RMK CA/MYD5WF
 9.RMK AUTOMATIC FARE QUOTE
10.FN/A/FCNY1700.00/SCNY0.00/RCNY1700.00/BCNY120.00/C3.00/XCNY120.00/
    TCNY50.00CN/TCNY70.00YQ/ACNY1820.00/P1
11.EI/MIANFEIGAIQI 免费改期/TUIPIAOSHOUFEI 退票收费/P1                -12.BJS734
```

通过航段参数查询：

1. 调用方法：

```
public PATResult doPatA(String location, String airline, String office, String group, String clFlag, Vector segments, String[] planeModes)
```

其中各参数的含义：

locationPNR 所在城市的三字码，标明属地，备用参数

airline 查询的航空公司，备用参数

office 运价可用的 office

group 团队标记，备用参数

clFlag 旅客类型，""表示成人，"CH"表示儿童，"IN"表示婴儿，"JC"表示因公带伤警察，"GM"表示伤残军人

segments 航段信息组，内部成员为 PNRAirSeg 对象

planeModes 航班机型组，内部成员为 String 的机型，和航段是一一对应

2. 查询结果说明：

PATA 查询结果存放于 PATAResult 中，其中各项对应如下：

运价信息： 同样，主机运价信息存放在 PATFareItem 中，通过 PATResult 中的 PATFareItem getFareItem(int index)取得

运价条数： PATFareItem 的条数通过 PATResult 中的 public int getFareNumber()获得

PATFareItem 说明：

获得某条运价的序号：`int getIndex()`
获得某条运价文本： `String getFare()`
获得 FC 对象：`(PNRFC)fareitem.getFc()`
获得 FN 对象：`(PNRFN)fareitem.getFn()`

3. 使用用例

```
try{
    PAT pat = new PAT();
    Vector segments = new Vector();
  PNRAirSeg seg = new PNRAirSeg("MU2705", 'H', "SHA", "PEK", "HK", 1,
            QDateTime.stringToDate("20100711 1000","YYYYMMDD HHMI"),
            QDateTime.stringToDate("20100711 1155","YYYYMMDD HHMI"),
            false, PNRAirSeg.AIRSEG_NORMAL);

  segments.add(seg);
  String[] planemodes = new String[]{"300"};

        PATResult result = pat.doPatA("PEK", "CA", "PEK099", "", "", segments, planemodes);
        System.out.println("PATA Result:");
        for(int i =0; i<result.farenumber; i++){
            PATFareItem item = (PATFareItem)result.getFareItem(i);
            System.out.println("FC:" + ((PNRFC)item.getFc()).makeString());
            System.out.println("FN:" + ((PNRFN)item.getFn()).makeString());
        }
}catch(Exception e){
    e.printStackTrace();
}
```

IBE 输出结果：

```
PATA Result:
FC:原文：SHA MU PEK 950.00H CNY950.00END
Segment:
        起始:SHA    承运人:MU    终到:PEK    运价：950.00    运价基础：H
总运价:CNY     950.00
FN:原文:
票面价：CNY950.00
实收：CNY950.00
代理费率：3.00
税项:          CN:CNY50.00         YQ:EXEMPT
```

6.2. 查询 PAT 公布运价

➢ eTerm PATA 查询过程

1.在主机中预定一个 PNR,然后进行 RT:

```
RT: W7H06
```

```
RT W7H06
 1.TEST/TEST W7H06
 2.  CA1831 Y   TU30MAR  PEKSHA HK1   0730 0940         E T3T2
 3.123456
 4.TL/0000/29MAR/PEK099
 5.SSR FOID CA HK1 NI123456789/P1
 6.SSR ADTK 1E BY PEK27MAR10/0730 OR CXL CA1831 Y30MAR
 7.RMK CA/L400E
 8.PEK099
```

2．执行 PAT:指令，得到运价：

```
PAT:
```

```
□FN FCNY 1130.00/ SCNY1130.00/ C3.00/ TCNY 50.00CN/ TCNY400.00YQ
FC PEK CA SHA 1130.00Y
-  CNY 1130.00 END
FP CASH, CNY
```

3．如果需要将此条运价录入到 PNR，则将整个运价文本作为指令 trans

```
 1.TEST/TEST W7H06
  2.  CA1831 Y   TU30MAR  PEKSHA HK1   0730 0940         E T3T2
  3.123456
  4.TL/0000/29MAR/PEK099
  5.FC/PEK CA SHA 1130.00Y CNY1130.00END
  6.SSR FOID CA HK1 NI123456789/P1
  7.SSR ADTK 1E BY PEK27MAR10/0730 OR CXL CA1831 Y30MAR
  8.RMK CA/L400E
  9.FN/FCNY1130.00/SCNY1130.00/C3.00/XCNY450.00/TCNY50.00CN/TCNY400.00YQ/
    ACNY1580.00
 10.FP/CASH,CNY
11.PEK099                                                  +
```

➢ IBE 的 PAT 接口使用过程

1. 调用方法：

和通过 PNR 查询 PAT:A 接口一样，方法是：

```
public PATResult doPat(String pnrno, String option, String type,int index,
        boolean ifinput)
```

其中各参数的含义：

pnr　　　PNR 编号

option　　查公布价则写””

type　　　查询类型，IN：婴儿；CH：儿童；JC：因公带伤警察；GM：伤残军人,不写则填 null 或""

index　　旅客序号,从 1 开始,查询净价时此项才有效

ifinput　　是否将查询的运价结果录入 PNR （默认公布运价查询结果只有一条）

2. 查询结果与主机的对应

```
□FN FCNY 1130.00/ SCNY1130.00/ C3.00/ TCNY 50.00CN/ TCNY400.00YQ
FC PEK CA SHA 1130.00Y
- CNY 1130.00 END
FP CASH, CNY
```

PATA 查询结果存放于 PATAResult 中，其中各项对应如下：

运价信息：　主机运价信息存放在 PATFareItem 中，通过 PATResult 中的 PATFareItem getFareItem(int index)取得

PATFareItem 说明：

获得某条运价的序号：int getIndex()

获得某条运价文本： String getFare()

运价条数：　PATFareItem 的条数通过 PATResult 中的 public int getFareNumber()获得

PATFareItem 说明：

获得某条运价的序号：int getIndex()

获得某条运价文本： String getFare()

获得 FC 对象：(PNRFC)fareitem.getFc()

获得 FN 对象：(PNRFN)fareitem.getFn()

3. 使用用例

```
PAT pat = new PAT();
try{
    PATResult    result = pat.doPat("ZQ803","",null,1,false);
    System.out.println(result);
      for(int i =0; i<result.farenumber; i++){
            PATFareItem item = (PATFareItem)result.getFareItem(i);
            System.out.println("FC:" + ((PNRFC)item.getFc()).makeString());
            System.out.println("FN:" + ((PNRFN)item.getFn()).makeString());
```

```
            }
        }catch(Exception e){
            e.printStackTrace();
        }
```

IBE 输出结果：

```
FC:原文：PEK CA SHA 1130.00Y
                                                    CNY1130.00END
Segment:
    起始:PEK   承运人:CA   终到:SHA   运价：1130.00   运价基础：Y
总运价:CNY    1130.00
FN:原文：
票面价：CNY1130.00
实收：CNY1130.00
代理费率：0.00
税项：       CN:CNY50.00      YQ:CNY400.00
```

7. 电子客票出票/废票

7.1. 电子客票的出票（ETDZ：）

- eTerm 电子客票出票过程：

eTerm 中运用主机指令出电子票如下：
预订 PNR 成功之后

```
 ETDZ:4              --①
ET PROCESSING…PLEASE WAIT!
ELETRONIC  TICKET ISSUED
```

IBE 的 ETDZ 在 com.travelsky.ibe.client.pnr.ETDZ 里实现，与 Eterm 的 ETDZ 不同是，需要将 PNR 和打票机号作为输入参数。

① 使用方法 issueTicket(String pnrno, int printerNo)。
pnrno：需要出票的 PNR 号码
printerNo：打票机号

出票成功后返回结果：

OK

- 电子客票出票举例
-

```
import com.travelsky.ibe.client.pnr.ETDZ;

public class TestETDZ {

	public static void main(String[] args) throws Exception {
		ETDZ etdz = new ETDZ();
		try {
			etdz.establishCtl(1);
			String prntRes = etdz.issueBSPTicket("PNRNO", 1);

			System.out.println(prntRes);
		} catch (Exception e) {
			// TODO 自动生成 catch 块
			e.printStackTrace();
		}

	}
}
```

TIPS：

1，假定是使用 1 号打票机。
2，PNRNO 表示 PNR 号码。
3，etdz.establishCtl(1);表示对 1 号打票机监控。

出错信息提示：

ETDZPretreatmentFailedException　　电子客票出票前的预处理过程失败
ETDZPosttreatmentFailedException　　电子客票出票后的后期处理过程失败
ETDZPartFailedException　　电子客票出票操作部分失败：对于超过 30 名旅客的 PNR，ETDZ 需要分次出票，每次为 30 名旅客出票，如果在 PNR 已经部分出票的情况下 ETDZ 指令报错，则将抛出此异常
ETDZNoPNRException　　作为请求参数的 PNR 不存在。

7.2. 提取电子客票记录

6.2.1 根据票号提取电子客票记录（DETR:TN/票号）

➢ eTerm 通过票号提取电子客票记录过程

eTerm 通过票号提取电子客票记录如下：

```
 DETR:TN/9999639898286                    --①
>DETR:TN/9999639898286
ISSUED BY: AIR CHINA          ORG/DST: BJS/CTU             BSP-D
TOUR CODE:                                           RECEIPT PRINTED
PASSENGER:  测试人
EXCH:                                  CONJ TKT:
O FM:1PEK CA      4116    F 16JUN 1500 OK F              40K OPEN FOR USE
        T3-- RL:KC920      /PRRXN 1E
   TO: CTU
FC: 16JUN09PEK CA CTU2160.00CNY2160.00END
FARE:                CNY 2160.00|FOP:CA3
TAX:                 CNY 50.00CN|OI:
TAX:                 CNYEXEMPTYQ|
TOTAL:                CNY 2210.00|TKTN: 999-9639898286
```

IBE 的提取电子客票记录功能在 com.travelsky.ibe.client.pnr.DETR 中实现，需要输入票号。

① 输入票号 TN ：对应 getTicketInfoByTktNo(String tn)中的 tn。

➢ IBE 通过票号提取电子客票记录

```
import com.travelsky.ibe.client.pnr.DETR;
import com.travelsky.ibe.client.pnr.DETRTKTResult;

public class TestDETR_1 {

	public static void main(String[] args) {
		DETR detr = new DETR();
		try {
			DETRTKTResult res = detr.getTicketInfoByTktNo("9999639898286");
			System.out.println(res.toString());
		} catch (Exception e) {
			System.out.println(e);
		}
	}
}
```

IBE 生成的 PNR 显示结果

```
ISSUED BY: AIR CHINA                    ORG/DST: BJS/CTU   ISI:   ETKT TYPE: BSP DOMESTIC
TOUR CODE:                                                   RECEIPT PRINTED
PASSENGER:  测试人             ADULT
EXCH:                                   CONJ TKT:
null FM:PEK TO CTU CA              4116 F      16JUN 1500      OK    F /    40K    OPEN FOR USE
              T3--  RL:KC920 /PRRXN    1E     BN:
FC: 16JUN09PEK CA CTU2160.00CNY2160.00END
FARE:                    CNY 2160.0 | FOP:CA3
TAX:                     CNY 50.0CN        | OI:
TOTAL:                   CNY 2210.0 | TKTN:999-9639898286
```

TIPS：

要确保输入的票号是真实票号

出错信息提示：

DETRInvalidTktNoException　　无法提取指定票号的 DETR 电子客票信息

6.2.2 根据证件号提取电子客票记录（DETR:NI/身份证号）

➢ eTerm 通过证件号提取电子客票记录过程

eTerm 通过证件号提取电子客票记录有以下两种情况：

第一种：此证件只有一张票面结果

```
 DETR:NI/110101197701012233                          --①
>DETR:TN/110101197701012233
ISSUED BY: AIR CHINA          ORG/DST: BJS/CTU                 BSP-D
TOUR CODE:                                                 RECEIPT PRINTED
PASSENGER:  测试人
EXCH:                                       CONJ TKT:
O FM:1PEK CA       4116    F 16JUN 1500 OK F                    40K OPEN FOR USE
         T3-- RL:KC920      /PRRXN 1E
   TO: CTU
FC: 16JUN09PEK CA CTU2160.00CNY2160.00END
FARE:                    CNY 2160.00|FOP:CA3
TAX:                     CNY 50.00CN|OI:
TAX:                     CNYEXEMPTYQ|
TOTAL:                    CNY 2210.00|TKTN: 999-9639898286
```

第二种：此证件有多张票面结果

```
 DETR:NI/110101197701012233                    --①
>DETR:NI/110101197701012233
>DETR:TN/999-9639898286                NAME: 测试人
    FOID:NI110101197701012233                  CA4116 /02AUG09/PEKCTU OPEN
>DETR:TN/999-9639898288                 NAME: 测试人
    FOID:NI110101197701012233                  CA4117 /25JUL09/CTUPEK OPEN
END OF SELECTION LIST
```

IBE 的提取电子客票记录功能在 com.travelsky.ibe.client.pnr.DETR 中实现，需要输入乘客的证件类型和证件号码。

① 输入票号 certcode：对应 getTicketInfoByCert(String certcode,String certno)中的 certcode,身份证请输入：NI，护照请输入：PP。

输入票号 certcode：对应 getTicketInfoByCert(String certcode,String certno)中的 certcode,身份证请输入：NI，护照请输入：PP。

➢ IBE 通过证件号提取电子客票记录

```
import java.util.Vector;

import com.travelsky.ibe.client.pnr.DETR;

public class TestDETR_1 {

	public static void main(String[] args) {
		DETR detr = new DETR();
		try {
			Vector result_vector = detr.getTicketInfoByCert("NI","");
		} catch (Exception e) {
			System.out.println(e);
		}
	}
}
```

IBE 生成的 PNR 显示结果：是一组 DETRTKTResult 组成的 Vector。

TIPS:

要确保输入的正确的证件类型和证件号

出错信息提示:

IBEException　　具体错误信息，详见此 Exception 中的返回信息。

6.2.3 根据旅客姓名提取电子客票记录

➢ eTerm 通过旅客姓名提取电子客票记录过程

eTerm 通过旅客姓名提取电子客票记录有以下两种情况：
第一种：此姓名只有一张票面结果

```
 DETR:NM/测试人                                --①
> DETR:NM/测试人
ISSUED BY: AIR CHINA          ORG/DST: BJS/CTU                BSP-D
TOUR CODE:                                            RECEIPT PRINTED
PASSENGER:  测试人
EXCH:                                   CONJ TKT:
O FM:1PEK CA      4116    F 16JUN 1500 OK F                40K OPEN FOR USE
       T3-- RL:KC920      /PRRXN 1E
   TO: CTU
FC: 16JUN09PEK CA CTU2160.00CNY2160.00END
FARE:                CNY 2160.00|FOP:CA3
TAX:                 CNY 50.00CN|OI:
TAX:                 CNYEXEMPTYQ|
TOTAL:               CNY 2210.00|TKTN: 999-9639898286
```

第二种：此姓名有多张票面结果

```
  DETR:NM/测试人                                --①
> DETR:NM/测试人
>DETR:TN/999-9639898286               NAME:  测试人
     FOID:NI110101197701012233              CA4116 /02AUG09/PEKCTU OPEN
>DETR:TN/999-9639898288               NAME:  测试人
     FOID:NI110101197701012233              CA4117 /25JUL09/CTUPEK OPEN
END OF SELECTION LIST
```

IBE 的提取电子客票记录功能在 com.travelsky.ibe.client.pnr.DETR 中实现，需要输入乘客的证件类型和证件号码。

① 输入旅客姓名 name：对应 getTicketInfoByName(String name,String flightno,Date deptime,boolean isOnlyOpen)中的 name。

② Flightno:此旅客所在的航班号，必须输入。

③ Deptime：航班起飞日期，必须输入。

④ Isopenonly:此变量为boolean类型，

输入 true 时，结果中全部票面为 open for use 状态；输入 false 时，则将所有票面都提出来，不论票面状态。

➢ IBE 通过旅客姓名提取电子客票记录

```
import java.util.Date;
import java.util.Vector;
import com.travelsky.ibe.client.pnr.DETR;

public class TestDETR_1 {

    public static void main(String[] args) {
        DETR detr = new DETR();
        try {
            Date flightDate = new Date("2009/08/01");
            Vector result_vector = detr.getTicketInfoByName("测试人","ca4116",flightDate,true);
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

IBE 生成的 PNR 显示结果：是一组 DETRTKTResult 组成的 Vector。

TIPS：

1， 要确保输入正确的旅客姓名。
2， 输入的航班时间，航班号是为了更快确认需要提出的票面，减少不必要的查找过程。

出错信息提示：

IBEException　具体错误信息，详见此 Exception 中的返回信息。

6.2.4 提取电子客票中旅客的证件信息（DETR:TN/票号,F）

➢ eTerm 通过票号提取电子客票旅客的身份识别号码

eTerm 通过票号提取电子客票旅客的身份识别号码如下：

```
 DETR:TN/9999639898286,F                    --①
>DETR:TN/880-2108078643,F
TKTN:8802108078643    NAME: 测试人
RCPT:
   1            NI110101197701012233
```

IBE 的提取电子客票记录功能在 com.travelsky.ibe.client.pnr.DETR 中实现，需要输入票号。

① 输入票号 TN：对应 getCredentialByTktNo(String tn)中的 tn。

➢ IBE 通过票号提取电子客票旅客的身份识别号码

```
import com.travelsky.ibe.client.pnr.DETR;
import com.travelsky.ibe.client.pnr.DETRFoidResult;

public class TestDETR_1 {

	public static void main(String[] args) {
		DETR detr = new DETR();
		try {
			DETRFoidResult res = detr.getCredentialByTktNo("880-2108078643");
			System.out.println(res);
		} catch (Exception e) {
			System.out.println(e);
		}
	}
}
```

IBE 生成的 PNR 显示结果

```
TKTN:8802108078643
NAME: 测试人
RCPT:
   1            NI110101197701012233
```

TIPS:

> 要确保输入的票号是真实票号

出错信息提示:

Exception　具体错误信息，详见此 Exception 中的返回信息。

7.3. 电子客票出票重试（ETRY:）

➢ eTerm 电子客票出票重试过程:

由于各种原因（比如系统传输等问题）导致没有提示出票成功或者出现了“ELECTRONIC TICKET TRANSACTION TIMEOUT”时，可以使用出票重试指令 ETRY，操作如下：

第一步：查看当天出票失败的 BSP-

〉TSL：4

```
*****************************************************************************
*                CAAC  MIS  OPTAT  DAILY-SALES-REPORT                       *
*                                                                           *
*   OFFICE : HAK128    IATA NUMBER : 08022766    DEVICE : 4/  63002         *
*   DATE   : 18AUG                               AIRLINE:   ALL             *
-----------------------------------------------------------------------------
  TKT-NUMBER       ORIG-DEST  COLLECTION   TAXS   COMM%      PNR   AGENT
-------------------------------------------------------------------------
880-5440200002    BJS CAN      1200.00       50.00  4.00     M4803    29821
880-5440200001    XIY BJS        10.00              4.00     S5405    29821
880-5440202032    XIY HAK     BSP ET ISSUE FAILED !!!!       R6D03    29817
*===========================================================================*
      TOTAL TICKETS:        3 (       0 TICKETS VOID /       0 TICKETS REFUND )
 ---------------NORMAL TICKETS-------------------------------------------
   NORMAL  FARE-- AMOUNT :        1210.00               CNY
      CARRIERS -- AMOUNT :        1161.60               CNY
        COMMIT -- AMOUNT :          48.40               CNY
   NORMAL  TAX -- AMOUNT :          50.00               CNY
 ---------------REFUND TICKETS-------------------------------------------
    NET REFUND -- AMOUNT :           0.00               CNY
     DEDUCTION -- AMOUNT :           0.00               CNY
 REFUND    TAX -- AMOUNT :           0.00               CNY
```

【说明】在销售报表中查看所有当前出票失败的记录，出票失败的记录在报表中以“BSP ET ISSUE FAILED”标示出来

第二步：提取出票失败的 BSP-ET PNR 记录

〉RT：R6D03

```
 1.测试人 R6D03
 2.  HU3068 Y   MO18OCT  XIYHAK HK1   2100 2340          E
 3.HAK/T HAK/T 0898-66701769/HAI NAN KAI SHENG INDUSTRY CO. LTD./ ABCDEFG
 4.NC
 5.T
 6.SSR FOID HU HK1 NI110101700101001/P1
 7.SSR OTHS 1E TKTL ADV TKT NBR TO HU BY 16OCT04/2100/PEK TIM/OR NO ALL SG/BCS
   HU3068 /Y/18OCT/XIYHAK
 8.SSR TKNE HU HK1 XIYHAK 3068 Y18OCT 8805440202032/1/DPN63002/P1
 9.RMK CA/BQH5E
10.FN/FCNY1730.00/SCNY1730.00/C3.00/XCNY50.00/TCNY50.00CN/ACNY1780.00
11.TN/880-5440202032/P1
12.FP/CASH,CNY
13.HAK128
```

【说明】出票失败的 PNR 记录中会自动生成 SSR TKNE 项和 RMK CA/XXXXX，缺少这两项的 PNR 将无法进行重试指令 ETRY。

第三步：执行重试指令 ETRY：

〉ETRY：

```
ET PROCESSING...PLEASE WAIT!    R6D03
ET TRANSACTION SUCCESS
```

第四步：查看销售报表中记录的状态

〉TSL：4

```
******************************************************************************
*                  CAAC  MIS  OPTAT  DAILY-SALES-REPORT                       *
*                                                                             *
*   OFFICE : HAK128     IATA NUMBER : 08022766     DEVICE : 4/  63002        *
*   DATE   : 18AUG                                 AIRLINE:   ALL            *
-------------------------------------------------------------------------------
  TKT-NUMBER      ORIG-DEST  COLLECTION   TAXS   COMM%      PNR    AGENT
---------------------------------------------------------------------------
880-5440200002    BJS CAN     1200.00      50.00  4.00     M4803   29821
880-5440200001    XIY BJS       10.00             4.00     S5405   29821
880-5440202032    XIY HAK     1730.00      50.00  4.00     R6D03   29817
*=============================================================================*
      TOTAL TICKETS:       3 (      0 TICKETS VOID /      0 TICKETS REFUND )
 ---------------NORMAL TICKETS--------------------------------------------------
   NORMAL  FARE-- AMOUNT :        2410.00              CNY
      CARRIERS -- AMOUNT :        2313.60              CNY
        COMMIT -- AMOUNT :          96.40              CNY
   NORMAL  TAX -- AMOUNT :          50.00              CNY
 ---------------REFUND TICKETS--------------------------------------------------
    NET REFUND -- AMOUNT :           0.00              CNY
     DEDUCTION -- AMOUNT :           0.00              CNY
 REFUND     TAX -- AMOUNT :          0.00              CNY
```

【说明】重试成功以后，销售报表中的原失败记录与其它成功出票的记录一样

IBE 的 ETRY 在 com.travelsky.ibe.client.pnr.ETDZ 里实现，与 Eterm 的 ETRY 不同是，需要将 PNR 作为输入参数。

① 使用方法 etryBSPTicket (String pnrno)。

pnrno：需要重新出票的 PNR 号码

重试出票成功后返回结果：

```
OK
```

电子客票出票举例

```
import com.travelsky.ibe.client.pnr.ETDZ;

public class TestETRY{

    public static void main(String[] args) throws Exception {
        ETDZ etdz = new ETDZ();
        try {
            etdz.establishCtl(1);
            String prntRes = etdz.etryBSPTicket ("PNRNO");
```

```
                System.out.println(prntRes);
            } catch (Exception e) {
                // TODO 自动生成 catch 块
                e.printStackTrace();
            }

        }
}
```

TIPS：

1，假定是使用 1 号打票机。
2，PNRNO 表示 PNR 号码。
3，etdz.establishCtl(1);表示对 1 号打票机监控。

出错信息提示：

CAN NOT RETRY THIS PNR. ETDZ FIRST　　该 PNR 没有执行 ETDZ 指令，不能执行重试操作

RMK CA/XXXXX NOT FOUND. PNR MISMATCH　没有返回 ICS PNR(航空公司大编码)，不能执行重试操作

PNR TICKETED 该 PNR 已被出票

CAN NOT RETRAY THIS ET, TICKET IS OUT OF DATE

该 ET 票不是当天销售的，不能执行重试操作

。

7.4. 电子客票废票（VT:）

➢ eTerm 电子客票废票过程:

eTerm 中运用主机指令作废电子票如下：

```
 VT:4/8805440202024            --①
ET TRANSACTION SUCCESS
```

IBE 的 VT 在 com.travelsky.ibe.client.pnr.VT 里实现，与 Eterm 的 ETDZ 不同是，需要将航空公司和 PNR 作为输入参数。

① 使用方法 voidETicket(String deviceNo, String ticketNo, String airline, String pnrNo)。

deviceNo：对应打票机号

ticketNo：对应要作废的票号

airline：航空公司代码，对于 BSP-ET 为"BSP"

pnrNo：pnrNo 订座系统纪录编号 Pnr，对于 BSP-ET 必须输入 CRS 系统的 PNR 编号，对于航空公司电子票则为""

废票成功后返回结果：

```
OK
```

- 电子客票废票举例
-

```
import com.travelsky.ibe.client.pnr.VT;

public class TestVT{

	public static void main(String[] args) throws Exception {
		VT vt = new VT();
		try {
			String prntRes = vt. voidETicket (4,"8805440202024"," ", "PNRNO");

			System.out.println(prntRes);
		} catch (Exception e) {
			// TODO 自动生成 catch 块
			e.printStackTrace();
		}

	}
}
```

TIPS：

1，假定是使用 4 号打票机。
2，PNRNO 表示 PNR 号码。

出错信息提示：

VTInvalidTktNoException	无法提取指定票号的电子客票信息
VTInvalidTktStatusException	电子客票的客票状态不正确
VTCannotEstCtrException	不能对打票机进行建立控制

7.5. 电子客票挂起（TSS:）

➢ eTerm 电子客票废票过程:

eTerm 中运用主机指令挂起电子票如下：

```
 Tss:tn/8805440202024/s             --①
ACCEPTED

DETR:TN/9992123815001
ISSUED BY: AIR CHINA                ORG/DST: PEK/SHA              ARL-D
TOUR CODE:                                               RECEIPT PRINTED
PASSENGER: CHONG/HSIAO WEI
EXCH:                          CONJ TKT:
O FM:1PEK CA    1501  F 06DEC 0830 OK F150                  40K SUSPENDED
    T3-- RL:J6HXL   /
  TO: SHA
FC: M  06DEC09PEK CA SHA1700.00CNY1700.00END
FARE:         CNY 1700.00|FOP:CC H1
TAX:          CNY 50.00CN|OI:
TAX:          CNY 50.00YQ|

Tss:tn/8805440202024/b              --②
ACCEPTED
DETR:TN/9992123815001
ISSUED BY: AIR CHINA                ORG/DST: PEK/SHA              ARL-D
TOUR CODE:                                               RECEIPT PRINTED
PASSENGER: CHONG/HSIAO WEI
EXCH:                          CONJ TKT:
O FM:1PEK CA    1501  F 06DEC 0830 OK F150                  40K OPEN FOR USE
    T3-- RL:J6HXL   /
  TO: SHA
FC: M  06DEC09PEK CA SHA1700.00CNY1700.00END
FARE:         CNY 1700.00|FOP:CC H1
TAX:          CNY 50.00CN|OI:
TAX:          CNY 50.00YQ|
```

IBE 的 TSS 在 com.travelsky.ibe.client.pnr.TSS 里实现，将需要挂起的票号作为为输入参数。

① 使用方法 SuspendByTktNo (String ticketNo)挂起。

ticketNo：对应要挂起的票号

挂起成功后返回结果：”OK”

② 使用方法 ResumeByTktNo (String ticketNo)解挂。

ticketNo：对应要挂起的票号

解挂成功后返回结果：”OK”

- 电子客票废票举例
-

```
            TSS tss = new TSS();
            try {
                //挂起
                 String Strsuspend = tss. SuspendByTktNo (”9992123815001”);
                System.out.println(Strsuspend);
               DETR detr =new DETR();

             //解挂
                  String StrRes = tss. ResumeByTktNo (”9992123815001”);
                System.out.println(StrRes);

            } catch (Exception e) {
                // TODO 自动生成 catch 块
                e.printStackTrace();
            }


        }
}
```

TIPS:

1， 也可以对票号对应的 PNR 挂起：SuspendByPNR(String pnrNo, Date issueDate)
解挂：ResumeByPNR(String pnrNo, Date issueDate)

出错信息提示:

TSSErrorException	无法挂起指定票号的电子客票

8. 电子客票非全屏模式生成修改打印退票单指令（TRFD:）

8.1. TRFD 指令创建/查看/打印退票单（非全屏模式）

➢ eTerm 电子客票创建退票单过程:

eTerm 中运用主机指令创建退票单如下:

```
>TRFU: M  4  / D/ 0
AIRLINE CODE       TKT NUMBER            -          CHECK
CONJUNCTION NO.  1  COUPON NO.  1 0000  2 0000  3 0000  4 0000
PASSENGER NAME
CURRENCY CODE  CNY-2  FORM OF PAYMENT   CASH
GROSS REFUND                                    ET-(Y/N):  Y
DEDUCTION                    COMMISSION       % =
TAX [ 1]  __ _________ [ 2]  __ _________ [ 3]  __ _________
    [ 4]  __ _________ [ 5]  __ _________ [ 6]  __ _________
    [ 7]  __ _________ [ 8]  __ _________ [ 9]  __ _________
    [10]  __ _________ [11]  __ _________ [12]  __ _________
    [13]  __ _________ [14]  __ _________ [15]  __ _________
    [16]  __ _________ [17]  __ _________ [18]  __ _________
    [19]  __ _________ [20]  __ _________ [21]  __ _________
    [22]  __ _________ [23]  __ _________ [24]  __ _________
    [25]  __ _________ [26]  __ _________ [27]  __ _________
REMARK                   CREDIT CARD
NET REFUND =            CNY
```

IBE 在 com.travelsky.ibe.client.pnr.TRFD 里实现。

使用方法 genRefundForm(String option, int prntNo, String tktType, String no)。

option:功能选项(AM:可以不输入任何票号创建退票记录;M:需要输入退票单号来创建查看退票记录;TM:查看退票记录需要输入 13 位票号;H:打印退票单,需要输入退票单号; TH:查看并打印退票记录,需要输入 13 位票号)

prntNo: 打印机编号

tktType: 国际/国内航班(I 为国际航班, D 为国内航班)

no: pnrNo 票号(可以为 13 位票号或者退票单号, 13 位票号格式为 999-1234567890),当生成退票单模版时写为 null

废票成功后返回 TRFDResult 结果, TRFDResult 作为下一步操作的必要输入参数。

生成退票模版举例

➢

```
import com.travelsky.ibe.client.pnr.TRFD;
import com.travelsky.ibe.client.pnr.TRFDResult;
public class TestTRFD{

    public static void main(String[] args) throws Exception {
        TRFD trfd = new TRFD();
        try {
            TRFDResult res = trfd. genRefundForm("am",4,"D",null);

            System.out.println(res);
        } catch (Exception e) {
            // TODO 自动生成 catch 块
            e.printStackTrace();
        }

    }
}
```

TIPS：

1，假定是使用 4 号打票机。
2，生成的 TRFDResult 中的内容，需要根据实际情况如实修改，作为最后提交退票单的依据。

出错信息提示:
IBEException: 具体错误信息，详见此 Exception 中的返回信息。

8.2. TRFD 提交退票单修改或删除退票单（非全屏模式）

IBE 在 com.travelsky.ibe.client.pnr.TRFD 里实现。

使用方法 changeRefundForm_new(TRFDResult trfdInfo, boolean isDel)。
trfdInfo：上一步得到的结果，在退票模板中填好的退票信息
isDel：删除或修改退票单(true 为删除,false 为修改)

操作成功后返回 String 类型结果，返回信息(<退票单号>：操作成功，"Error : <错误信息>"：操作失败)。
修改退票单举例
➢

```
import com.travelsky.ibe.client.pnr.TRFD;
public class TestTRFD{

    public static void main(String[] args) throws Exception {
        TRFD trfd = new TRFD();
        try {
```

```
            String res= trfd. changeRefundForm_new(trfdinfo,false);

            System.out.println(res);
        } catch (Exception e) {
            // TODO  自动生成  catch  块
            e.printStackTrace();
        }

    }
}
```

TIPS：

1，trfdinfo:表示使用上一步生成的退票单模版，填写完其中相关信息后的结果
2，提交的退票单信息要如实填写。

出错信息提示： IBEException：具体错误信息，详见此 Exception 中的返回信息。

9. 国际票查询

9.1. 通过 PNR 显示公布运价结果 QTE

实现类：com.travelsky.ibe.client.QTE

- eTerm 查询 QTE 过程

1. 订取一个国际航段的 PNR，然后做 RT 操作，例如：

```
RT:JNPKY
```

```
1.测试 JNPKY
 2.  CA8888 C   MO20JUL  PEKLAX HK1   1200 1212     E T3--OP-UA888
 3.123456
 4.TL/0000/29JUN/PEK888
 5.SSR FOID CA HK1 NI1234567890/P1
 6.SSR ADTK CA BY PEK17JUL09/1200 OR CXL CA8888 C20JUL
 7.RMK AK HDQUA QQX926 - CA CS
 8.PEK888
```

2. 使用 QTE 指令查询运价

```
QTE:
```

```
FSI/
S CA  8888C20JUL PEK1200 1212LAX0S
01 CXUAO             26322 CNY                 INCL TAX
*ATTN PRICED ON 04JUN09*1014
 BJS
 LAX CXUAO          NVB     NVA20JUL PC
FARE  CNY   26000
TAX   CNY     90CN CNY    111US CNY    121XT
TOTAL CNY   26322
20JUL09BJS CA LAX3800.43NUC3800.43END ROE6.841320
XT CNY 35XA CNY 48XY CNY 38YC
ENDOS REF-CNY100
```

- IBE 的 QTE 接口使用过程

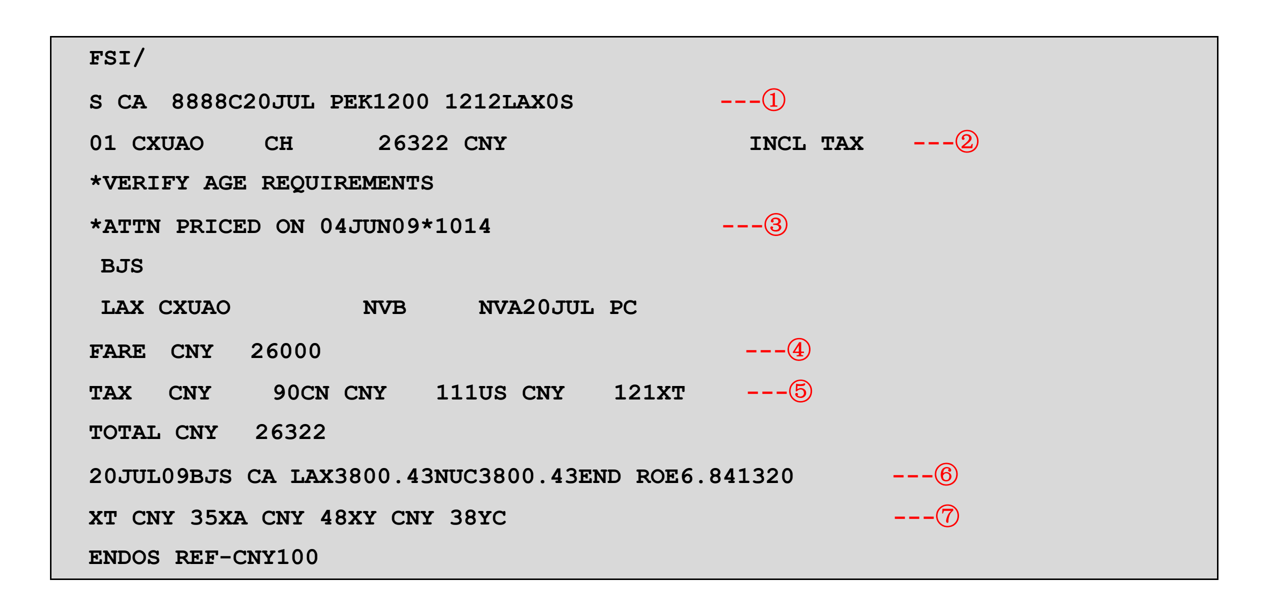
IBE 的 QTE 接口使用的相关类为：
com.travelsky.ibe.client.QTE
com.travelsky.ibe.client.XSFSPResult
com.travelsky.ibe.client.XSFSPFare
com.travelsky.ibe.client.XSFSQResult

1. QTE 接口调用方法：

```
public XSFSPResult doQTE(String pnr,String querytype,String faretype,String currency,boolean lowestonly,String validateCarrier,boolean isApplied,String index)
```

其中各参数的含义：
pnr PNR 编号
querytype 运价类型，如"CH",不指定为""或 null
faretype 查询类型，正常票价为"N"或特殊票价为"S",不指定可使用 null 或""
currency 需要计算的货币类型
lowestonly 是否仅查询给出的运价折扣种类中的最低运价
validateCarrier 出票航空公司
isApplied 是否使用代开协议
index 如果查询的结果多于一条，则输入"1"或"2"序号以查询税项明细，查询结果保存在 XSFSQResult 中,输入“NONE”对所有运价均不进行明细查询。

2. QTE 查询结果与主机的对应



```
FSI/
S CA  8888C20JUL PEK1200 1212LAX0S          ---①
01 CXUAO     CH       26322 CNY                INCL TAX     ---②
*VERIFY AGE REQUIREMENTS
*ATTN PRICED ON 04JUN09*1014                ---③
 BJS
 LAX CXUAO           NVB      NVA20JUL PC
FARE  CNY   26000                            ---④
TAX   CNY     90CN CNY    111US CNY    121XT  ---⑤
TOTAL CNY   26322
20JUL09BJS CA LAX3800.43NUC3800.43END ROE6.841320       ---⑥
XT CNY 35XA CNY 48XY CNY 38YC                          ---⑦
ENDOS REF-CNY100
```

① 航段相关信息：返回主机原文，如“S CA 8888C20JUL PEK1200 1212LAX0S”

调用方法：XSFSPResult 中的 String getSegAt(int i)，i 为航段信息的序号

② 基本运价信息：

包含在 XSFSPResult 的 XSFSPFare 项中，可以通过 getFare(int i)方法来获得。

1.运价编号：如图中的“01”，

调用方法：XSFSPFare 中的 String getIndex()

2.基础运价代码：如图中的 “CXUA0”，

调用方法：XSFSPFare 中的 String getFarebasis ()

3.基础运价代码：如果查询儿童票价，则会有“CH”

调用方法：XSFSPFare 中的 String getFlag()

4.含税的运价金额：如图中的“26322”

调用方法：XSFSPFare 中的 double getFare()

5.货币类型：如图中的“CNY”

调用方法：XSFSPFare 中的 String getCurrency()

6.税项说明信息：如图中的“INCL TAX”

调用方法：XSFSPFare 中的 String getTAXinfo()

其中④⑤⑥⑦为运价详细信息，存放在 XSFSPFare 中的 fsq 里，通过调用 XSFSQResult getFsq()方法获得，下面进行详细描述。

③ 额外信息：返回主机原文，如“*VERIFY AGE REQUIREMENTS”，“*ATTN PRICED ON 04JUN09*1014”

调用方法： XSFSPResult 中的 String getRemarkAt(int i)，i 为 Remark 对象的序号

④ 不含税运价金额：如图中的“FARE CNY 26000”

调用方法：运价金额使用 XSFSQResult 的 double getFare()方法，运价货币类型使用 XSFSQResult 的 String getCurrency()方法

⑤ 税项明细：如图中的“TAX CNY 90CN CNY 111US CNY 121XT”，其中 XT 税的明细在⑦中，IBE 返回的税项代码是包含 XT 明细的，而不会单独列出 XT 税。

调用方法：税款数量：XSFSQResult 的 int getTaxCount()

指定的税款金额：XSFSQResult 的 double getTax(int i)，i 为税款序号

指定的税款货币类型：XSFSQResult 的 String getTaxCurrency(int i)，i 为指定的索引序号

指定的税款代码：XSFSQResult 的 String getTaxCode(int i),i 为指定的索引序号

⑥ FC 项信息：如主机中的“20JUL09BJS CA LAX3800.43NUC3800.43END ROE6.841320”

调用方法： XSFSQResult 的 double getFc()方法

- IBE 的 XS FSP 接口使用用例

1.查询 PNR 的成人运价，指定出票航空公司和货币代码，使用代开协议

```
/*
*查询PNR 的成人运价，指定出票航空公司和货币代码，使用代开协议
* */
  QTE qte = new QTE();
  XSFSPResult result = qte.doQTE("JNPKY","",null,"CNY",false,"CA",true,"1");
  System.out.println(result.toString());
```

主机查询结果：

```
FSI/CA///CNY
S CA  8888C20JUL PEK1200 1212LAX0S
01 CXUAO             26850 CNY                  INCL TAX
*ATTN PRICED ON 17JUN09*0934
 BJS
 LAX CXUAO          NVB     NVA20JUL PC
FARE  CNY   26000
TAX   CNY     90CN CNY    111US CNY    649XT
TOTAL CNY   26850
20JUL09BJS CA LAX3800.43NUC3800.43END ROE6.841320
XT CNY 35XA CNY 48XY CNY 38YC CNY 28YQ CNY 500YR
ENDOS REF-CNY100
```

IBE 生成的 QTE 查询结果

```
S CA    8888C20JUL PEK1200 1212LAX0S

01    $ CXUAO              0026850.00       CNY  INCL TAX
FARE:CNY26000.00
NUC: 3800.43
ROE: 6.84132
TAX:
        CNY  90.00CN
        CNY  111.00US
        CNY  35.00XA
        CNY  48.00XY
        CNY  38.00YC
        CNY  28.00YQ
        CNY  500.00YR

*ATTN PRICED ON 09JUN09*1706
```

2. 查询 PNR 的儿童运价，指定出票航空公司和货币代码，使用代开协议

```
/*
*  查询PNR 的儿童运价，指定出票航空公司和货币代码，使用代开协议
* */
   QTE qte = new QTE();
   XSFSPResult result = qte.doQTE("JNPKY","CH",null,"CNY",false,"CA",true,"ALL");
   System.out.println(result.toString());
```

主机查询结果：

```
FSICH/CA///CNY
S CA  8888C20JUL PEK1200 1212LAX0S
01 CXUAO     CH       20260 CNY                  INCL TAX
*VERIFY AGE REQUIREMENTS
*ATTN PRICED ON 17JUN09*0940
 BJS
 LAX CXUAO    CH     NVB      NVA20JUL PC
FARE  CNY   19500
TAX   CNY    111US CNY     35XA CNY    614XT
TOTAL CNY   20260
20JUL09BJS CA LAX2850.32NUC2850.32END ROE6.841320
XT CNY 48XY CNY 38YC CNY 28YQ CNY 500YR
ENDOS REF-CNY100
```

IBE 生成的 QTE 查询结果

```
S CA   8888C20JUL PEK1200 1212LAX0S

01    $ CXUAO        CH   0020260.00       CNY INCL TAX
FARE:CNY19500.00
NUC: 2850.32
ROE: 6.84132
TAX:
       CNY 111.00US
       CNY 35.00XA
       CNY 48.00XY
       CNY 38.00YC
       CNY 28.00YQ
       CNY 500.00YR

*VERIFY AGE REQUIREMENTS
```

*ATTN PRICED ON 09JUN09*1715*

TIPS：

请求中的出票航空公司参数 validateCarrier 设置与否，会关系到查询结果的税项种类。
QTE 接口查询结果不会包括 XT 税项，而是返回 XT 税的明细税项

9.2. 查询汇率 XS FSC

实现类：com.travelsky.ibe.client.XSFSC

- eTerm 查询 XS FSC 过程

```
XS FSC USD*TYO/10JUN09
```

主机返回结果：

```
FSC USD/JPY/10JUN09
RATE BSR 1USD=99.61JPY
RATE BBR 1USD=97.61JPY
RATE ICH 1USD=95.182JPY
PAGE 1/1
```

- IBE 的 QTE 接口使用过程

  IBE 的 QTE 接口使用的相关类为：
  com.travelsky.ibe.client.XSFSC
  com.travelsky.ibe.client.XSFSCRequest
  com.travelsky.ibe.client.XSFSCRates

  1．FSC 接口调用方法：

```
public XSFSCResult fsc(XSFSCRequest req) throws Exception
```

```
XS FSC USD*TYO/10JUN09
```

其中参数信息放在 XSFSCRequest 中

setOCurrency(String currency)

设置原始货币代码，如图中的“USD”

setCityCode(String cityCode)

设置需要兑换到哪个城市/机场所在国货币，如图中的“JPY”

setDate(java.util.Date date)

设置查哪天的汇率，如图中的“10JUN09”

2. FSC 查询结果与主机的对应：

```
FSC USD/JPY/10JUN09                ---①
RATE BSR 1USD=99.61JPY             ---②
RATE BBR 1USD=97.61JPY             ---③
RATE ICH 1USD=95.182JPY            ---④
PAGE 1/1
```

因 XSFSCResult 实现了 XSFSCRates，所以返回结果都使用 XSFSCRates

① 获得原始货币代码和换算的目标货币代码：

1.获得原始货币代码：如图中“USD”，

调用方法：

XSFSCRates 中的 String getOrginCurrency()

2.获得换算的目标货币代码：如图中 “JPY”

调用方法：

XSFSCRates 中的 String getExchangedCurrency()

② 获得 BSR 汇率：

1.获得原始货币代码，如图中“USD”，

调用方法：

XSFSCRates 中的 String getOrginCurrency(XSFSCRates.BSR)

2.获得换算的目标货币代码，如图中“JPY”

调用方法：

XSFSCRates 中的 String getExchangedCurrency(XSFSCRates.BSR)

3.获得货币转换比率中的原货币数量，如图中“1”USD

调用方法：

XSFSCRates 中的 double getRateOriginVal(XSFSCRates.BSR)

4.获得货币转换比率中的换算后货币数量，如图中“99.61”USD

调用方法：

XSFSCRates 中的 double getRateExchangedVal(XSFSCRates.BSR)

③ 获得 BBR 汇率：

1.获得原始货币代码，如“USD”，

调用方法：

XSFSCRates 中的 String getOrginCurrency(XSFSCRates.BSR)

2.获得换算的目标货币代码，如图中“JPY”

调用方法：

XSFSCRates 中的 String getExchangedCurrency(XSFSCRates.BSR)

3.获得货币转换比率中的原货币数量，如图中“1”USD

调用方法：

XSFSCRates 中的 double getRateOriginVal(XSFSCRates.BSR)

4.获得货币转换比率中的换算后货币数量，如图中“97.61”USD

调用方法：

XSFSCRates 中的 double getRateExchangedVal(XSFSCRates.BSR)

④ 获得 ICH 汇率：

1.获得原始货币代码，如图中“USD”，

调用方法：

XSFSCRates 中的 String getOrginCurrency(XSFSCRates.ICH)

2.获得换算的目标货币代码，如图中“JPY”

调用方法：

XSFSCRates 中的 String getExchangedCurrency(XSFSCRates.ICH)

3.获得货币转换比率中的原货币数量，如图中“1”USD

调用方法：

XSFSCRates 中的 double getRateOriginVal(XSFSCRates.ICH)

4.获得货币转换比率中的换算后货币数量，如图中“95.182”USD

调用方法：

XSFSCRates 中的 double getRateExchangedVal(XSFSCRates.ICH)

- IBE 实现 XS FSC 汇率查询用例

查询美元到日元的汇率

```
/*
         * 成人单人单航段预定示例
         * */
          XSFSC fsc = new XSFSC();
          XSFSCRequest request = new XSFSCRequest();
          request.setCityCode("TYO");
          request.setDate(new Date());
          request.setOCurrency("USD");
          XSFSCRates result = fsc.fsc(request);
          System.out.println(result.toString());
```

主机查询结果：

```
FSC USD/JPY/10JUN09
RATE BSR 1USD=99.61JPY
RATE BBR 1USD=97.61JPY
RATE ICH 1USD=95.182JPY
PAGE 1/1
```

IBE 生成的查询结果

Inquery parameters:USD/JPY/11JUN09
RATE BBR 1.000000USD = 97.610000JPY
RATE BSR 1.000000USD = 99.610000JPY
RATE ICH 1.000000USD = 95.182000JPY

备注信息:

原文:
FSC USD/JPY/11JUN09
RATE BSR 1USD=99.61JPY
RATE BBR 1USD=97.61JPY
RATE ICH 1USD=95.182JPY
PAGE 1/1

9.3. 国际运价计算 XS FSP

实现类：com.travelsky.ibe.client.XSFSP

- eTerm 查询 XS FSP 过程

```
XS FSP PEK 25jun #Cca FRA#R/CA///cny
```

主机返回结果：

```
FSI/CA///CNY
U*CA       25JUN PEK         FRA0S
01 FOWCN              47668 CNY      RBFL         INCL TAX
02 FOWLH              47538 CNY      RBFL         INCL TAX
03 COW                41868 CNY      RB           INCL TAX
04 YOW                24148 CNY      RB           INCL TAX
05 DOWCN              22968 CNY      RBFL         INCL TAX
06 EKGVO               8228 CNY      RBFL         INCL TAX
07 KKAPOCN1            5548 CNY      RBFL         INCL TAX
*1,2,5,6,7*FLIGHT RESTRICTIONS APPLY
*ATTN PRICED ON 16JUN09*1404
PAGE 1/1
```

查询某一条运价的详细信息：

```
XS FSQ 1
```

```
FSI/CA///CNY
U*CA      25JUN PEK        FRA0S
01 FOWCN             47668 CNY     RBFL        INCL TAX
02 FOWLH             47538 CNY     RBFL        INCL TAX
03 COW               41868 CNY     RB          INCL TAX
04 YOW               24148 CNY     RB          INCL TAX
05 DOWCN             22968 CNY     RBFL        INCL TAX
06 EKGVO              8228 CNY     RBFL        INCL TAX
07 KKAPOCN1           5548 CNY     RBFL        INCL TAX
*1,2,5,6,7*FLIGHT RESTRICTIONS APPLY
*ATTN PRICED ON 16JUN09*1404
 BJS
 FRA FOWCN         NVB     NVA     40K
FARE  CNY   46800
TAX   CNY     90CN CNY     28YQ CNY    750YR
TOTAL CNY   47668
25JUN09BJS CA FRA6840.78NUC6840.78END ROE6.841320
ENDOS NON-END/PENALTY APPLS
```

- IBE 的 XS FSP 接口使用过程

IBE 的 XS FSP 接口使用的相关类为:
com.travelsky.ibe.client.XSFSP
com.travelsky.ibe.client.XSFSPResult
com.travelsky.ibe.client.XSFSPInformation
com.travelsky.ibe.client.XSFSPSegmentReq
com.travelsky.ibe.client.XSFSQResult

1. XS FSP 接口调用方法:

```
public XSFSPResult FSP(XSFSPInformation fspinfo, String fsqindex)throws IBEException
```

```
XS FSP CTU 27jun #Cca PEK 28jun #Cca FRA 30jun #Cca PEK#R////cny
```

接口各参数含义:
fspinfo 参数集合,包含起始机场,航段参数 XSFSPSemgentReq 集,运价种类,查询类型
fsqindex 某条运价的索引号，用于查询该运价的明细

其中 XSFSPInformation 是请求参数的集合，其中各参数使用方法：

XSFSPInformation——

setFaretype(String faretype):

设置起始机场三字码(必输项)

addSegseq(XSFSPSegmentReq seg):

添加航段参数，具体说明见下面的 XSFSPSegmentReq

setFlag(String flag):

运价种类,如(CH),可为空(null/"")，此时查询通用票价

setFaretype(String faretype):

查询类型,正常票价为"N"或特殊票价为"S",不选择可使用 null/""

setCurrencyCity(String currency):

设置需要计算的货币类型，如主机中的“CNY”

setLowestonly(boolean lowestonly):

设置是否仅查询给出的运价折扣种类中的最低运价

setValidateCarrier(String validateCarrier):

设置出票航空公司

XSFSPSegmentReq——

setDstcity(String dstcity):

设置航段到达机场三字码（必输项）

setCarrier(String carrier):

设置承运人代码,不设置或设为 null 或空串则表示不指定，如主机请求里“#Cca”中的 ca

setDate(String date):

设置日期格式(DDMMM),不设置或设为 null 或空串则表示不指定，如主机中的“27jun”,“28jun”

setTraveldirect(String traveldirect)

设置旅行方向(EH,PA,AT,AP,SA,TS,FE,SP,PO),不设置或设置为空串则表示不指定

setStopoverflag(String stopoverflag)

设置经停点标志(“/”为经停点,“-”为非经停点),在几个航段信息中不能同时存在这两个符号

可不写（赋值为 null/""）

setCabin(String Cabin)

设置查询指定舱位,不设置或设置为 null 或空串表示查询时不指定舱位

2. XS FSP 查询结果与主机的对应：

```
XS FSP PEK 25jun #Cca FRA#R/CA///cny
```

```
FSI/CA///CNY
U*CA       25JUN PEK         FRA0S                                      ---①
01 FOWCN                  47668 CNY      RBFL          INCL TAX         ---②
02 FOWLH                  47538 CNY      RBFL          INCL TAX
03 COW                    41868 CNY      RB            INCL TAX
04 YOW                    24148 CNY      RB            INCL TAX
05 DOWCN                  22968 CNY      RBFL          INCL TAX
06 EKGVO                   8228 CNY      RBFL          INCL TAX
07 KKAPOCN1                5548 CNY      RBFL          INCL TAX
*1,2,5,6,7*FLIGHT RESTRICTIONS APPLY
*ATTN PRICED ON 16JUN09*1404                                            ---③
PAGE 1/1
```

```
XS FSQ 1
```

```
FSI/CA///CNY
U*CA       25JUN PEK         FRA0S
01 FOWCN                  47668 CNY      RBFL          INCL TAX
02 FOWLH                  47538 CNY      RBFL          INCL TAX
03 COW                    41868 CNY      RB            INCL TAX
04 YOW                    24148 CNY      RB            INCL TAX
05 DOWCN                  22968 CNY      RBFL          INCL TAX
06 EKGVO                   8228 CNY      RBFL          INCL TAX
07 KKAPOCN1                5548 CNY      RBFL          INCL TAX
*1,2,5,6,7*FLIGHT RESTRICTIONS APPLY
*ATTN PRICED ON 16JUN09*1404
 BJS
 FRA FOWCN            NVB       NVA       40K
FARE  CNY   46800                                                       ---④
TAX   CNY     90CN CNY     28YQ CNY    750YR                            ---⑤
TOTAL CNY   47668
25JUN09BJS CA FRA6840.78NUC6840.78END ROE6.841320                       ---⑥
ENDOS NON-END/PENALTY APPLS
```

① 航段相关信息：返回主机原文，如“U*CA   25JUN PEK   FRA0S”

调用方法：XSFSPResult 中的 String getSegAt(int i)，i 为航段信息的序号

② 基本运价信息：

包含在 XSFSPResult 的 XSFSPFare 项中，可以通过 getFare(int i)方法来获得。

1.运价编号：如图中的“01”，

调用方法：XSFSPFare 中的 String getIndex()

2.基础运价代码：如图中的“FOWCN”，

调用方法：XSFSPFare 中的 String getFarebasis ()
3.运价种类：如果查询儿童票价，则会有“CH”
调用方法：XSFSPFare 中的 String getFlag()
4.含税的运价金额：如图中的“47668”
调用方法：XSFSPFare 中的 double getFare()
5.货币类型：如主机中的“CNY”
调用方法：XSFSPFare 中的 String getCurrency()
6.税项说明信息：如主机中的“INCL TAX”
调用方法：XSFSPFare 中的 String getTAXinfo()
其中④⑤⑥为运价详细信息，存放在 XSFSPFare 中的 fsq 里，只有在请求里指定 FSQ 序号方可得到。通过调用 XSFSQResult getFsq()方法获得，下面进行详细描述。

③ 额外信息：返回主机原文，如“*1,2,5,6,7*FLIGHT RESTRICTIONS APPLY”,“*ATTN PRICED ON 16JUN09*1404”
调用方法： XSFSPResult 中的 String getRemarkAt(int i)，i 为 Remark 对象的序号

④ 不含税运价金额：如主机中的“FARE CNY 46800”
调用方法：运价金额使用 XSFSQResult 的 double getFare()方法，运价货币类型使用 XSFSQResult 的 String getCurrency()方法

⑤ 税项明细：如主机中的“TAX CNY 90CN CNY 28YQ CNY 750YR”，
调用方法：
税款数量：XSFSQResult 的 int getTaxCount()
指定的税款金额：XSFSQResult 的 double getTax(int i)，i 为税款序号
指定的税款货币类型：XSFSQResult 的 String getTaxCurrency(int i)，i 为指定的索引序号
指定的税款代码：XSFSQResult 的 String getTaxCode(int i),i 为指定的索引序号

⑥ FC 项信息：如主机中的“25JUN09BJS CA FRA6840.78NUC6840.78END ROE6.841320”
调用方法： XSFSQResult 的 double getFc()方法

➢ IBE 的 XS FSP 接口使用用例

1. 查询单航段成人运价，指定出票航空公司
主机请求：

```
XS FSP PEK 25jun #Cca FRA#R/CA///cny
```

```
/*
 * 查询单航段成人运价，指定出票航空公司
 * */
XSFSP fsp=new XSFSP();

XSFSPInformation info=new XSFSPInformation();

info.setFlag(null);
//起始机场
info.setOrgcity("PEK");
//票价类型,普通运价为"N"
info.setFaretype("");
//指定货币类型
info.setCurrencyCity("cny");
//是否求最低运价
info.setLowestonly(false);

//第一航段
XSFSPSegmentReq seg1=new XSFSPSegmentReq();
//到达机场
seg1.setDstcity("FRA");
//起飞时间
seg1.setDate("25jun");
//承运人
seg1.setCarrier("ca");
//添加第一航段
info.addSegseq(seg1);
//设置出票航空公司
info.setValidateCarrier("CA");
//调用 FSP 方法，并查询第一条运价明细
XSFSPResult xr=fsp.FSP(info,"1");

System.out.println(xr);
```

主机查询结果：

XS FSP 查询结果：

```
FSI/CA///CNY
U*CA      25JUN PEK        FRA0S
01 FOWCN              47668 CNY      RBFL         INCL TAX
02 FOWLH              47538 CNY      RBFL         INCL TAX
03 COW                41868 CNY      RB           INCL TAX
04 YOW                24148 CNY      RB           INCL TAX
```

```
05 DOWCN              22968 CNY     RBFL         INCL TAX
06 EKGVO               8228 CNY     RBFL         INCL TAX
07 KKAPOCN1            5548 CNY     RBFL         INCL TAX
*1,2,5,6,7*FLIGHT RESTRICTIONS APPLY
*ATTN PRICED ON 16JUN09*1404
PAGE 1/1
```

FSQ 查询结果:

```
XS FSQ 1
```

```
FSI/CA///CNY
U*CA      25JUN PEK        FRA0S
01 FOWCN              47668 CNY     RBFL         INCL TAX
02 FOWLH              47538 CNY     RBFL         INCL TAX
03 COW                41868 CNY     RB           INCL TAX
04 YOW                24148 CNY     RB           INCL TAX
05 DOWCN              22968 CNY     RBFL         INCL TAX
06 EKGVO               8228 CNY     RBFL         INCL TAX
07 KKAPOCN1            5548 CNY     RBFL         INCL TAX
*1,2,5,6,7*FLIGHT RESTRICTIONS APPLY
*ATTN PRICED ON 16JUN09*1404
 BJS
 FRA FOWCN         NVB     NVA     40K
FARE  CNY   46800
TAX   CNY    90CN CNY    28YQ CNY    750YR
TOTAL CNY   47668
25JUN09BJS CA FRA6840.78NUC6840.78END ROE6.841320
ENDOS NON-END/PENALTY APPLS
```

IBE 的查询结果:

```
U*CA        25JUN PEK          FRA0S

01   $ FOWCN              0047668.00       CNY  INCL TAX
FARE:CNY46800.00
NUC: 6840.78
ROE: 6.84132
TAX:

FC:
25JUN09BJS CA FRA6840.78NUC6840.78END ROE6.841320
原文：25JUN09BJS CA FRA 6840.78 NUC6840.78END ROE6.841320
FC 日期：25JUN09
Segment:
          起始:BJS    承运人:CA    终到:FRA    运价：6840.78
总运价:NUC     6840.78    ROE:6.841320     CNY  90.00CN
      CNY  28.00YQ
      CNY  750.00YR

02   $ FOWLH              0047538.00       CNY  INCL TAX
03   $ COW           0041868.00       CNY  INCL TAX
04   $ YOW           0024148.00       CNY  INCL TAX
05   $ DOWCN              0022968.00       CNY  INCL TAX
06   $ EKGVO              0008228.00       CNY  INCL TAX
07   $ KKAPOCN1           0005548.00       CNY  INCL TAX
*1,2,5,6,7*FLIGHT RESTRICTIONS APPLY

*ATTN PRICED ON 16JUN09*1404
```

2．查询多航段有缺口程的成人运价，指定出票航空公司

主机请求：

```
XS FSP PEK 25jun #Cca FRA // CTU 30jun #Cca PEK#R/CA///cny
```

```
/*
* 查询多航段有缺口程的成人运价，指定出票航空公司
* */
XSFSP fsp=new XSFSP();

XSFSPInformation info=new XSFSPInformation();
```

```
        info.setFlag(null);
        //起始机场
        info.setOrgcity("PEK");
        //票价类型,普通运价为"N"
        info.setFaretype("");
        //指定货币类型
        info.setCurrencyCity("cny");
        //是否求最低运价
        info.setLowestonly(false);

        //第一航段
        XSFSPSegmentReq seg1=new XSFSPSegmentReq();
        //到达机场
        seg1.setDstcity("FRA");
        //起飞时间
        seg1.setDate("25jun");
        //承运人
        seg1.setCarrier("ca");
        //添加第一航段
        info.addSegseq(seg1);

        //第二机场
        XSFSPSegmentReq seg2=new XSFSPSegmentReq();
        //到达机场
        seg2.setDstcity("CTU");
        seg2.setDate("26jun");
        //设置缺口程，将对应航段的承运人参数设置为"//"
        seg2.setCarrier("//");
        info.addSegseq(seg2);

        //第三机场
        XSFSPSegmentReq seg3=new XSFSPSegmentReq();
        //到达机场
        seg3.setDstcity("PEK");
        seg3.setDate("30jun");
        //承运人
        seg3.setCarrier("ca");
        info.addSegseq(seg3);

        //设置出票航空公司
        info.setValidateCarrier("CA");          //调用 FSP 方法，并查询第一条运价明细
        XSFSPResult xr=fsp.FSP(info,"1");

       System.out.println(xr);
```

主机查询结果：

XS FSP 查询结果：

```
FSI/CA///CNY
U*CA      25JUN PEK        FRA0U
U*CA      30JUN CTU        PEK0S
01 FRT+FIF          91378 CNY     RB          INCL TAX
02 CRT+CIF          58698 CNY     RB          INCL TAX
03 YRT+YIF          37878 CNY     RB          INCL TAX
04 EKGV+YIF         25638 CNY     RBFL        INCL TAX
05 DOWCN+C          24888 CNY     RBFL        INCL TAX
06 QPR6MA+YIF       22398 CNY     RBFL        INCL TAX
07 EKGVO+Y           9718 CNY     RBFL        INCL TAX
08 KKAPOCN1+Y        7038 CNY     RBFL        INCL TAX
*4,5,6,7,8*FLIGHT RESTRICTIONS APPLY
*5,6,7,8*TKT STOCK RESTR
*ATTN CHANGE TO NO STOPOVER MAY VARY FARE
*ATTN PRICED ON 16JUN09*1605
PAGE 1/1
```

FSQ 查询结果：

```
XS FSQ 1
```

```
BJS
 FRA FRT           NVB     NVA     40K
 CTU     S U R F A C E
 BJS FIF           NVB     NVA     40K
FARE  CNY   90460
TAX   CNY    140CN CNY     28YQ CNY    750YR
TOTAL CNY   91378
25JUN09BJS CA FRA7308.53//CTU CA BJS M5914.06NUC13222.59END
ROE6.841320
ATTN* CHANGE TO NO STOPOVER MAY VARY FARE
```

IBE 的查询结果：

*U*CA 25JUN PEK FRA0U*

*U*CA 30JUN CTU PEK0S*

01 $ FRT+FIF 0091378.00 CNY INCL TAX
FARE:CNY90460.00
NUC: 13222.59
ROE: 6.84132
TAX:

FC:
25JUN09BJS CA FRA7308.53//CTU CA BJS M5914.06NUC13222.59END ROE6.841320
原文：25JUN09BJS CA FRA 7308.53 // CTU CA BJS M 5914.06 NUC13222.59END ROE6.841320
FC 日期：25JUN09
Segment:
起始:BJS 承运人:CA 终到:FRA 运价：7308.53
Segment:
起始:FRA 承运人:// 终到:CTU
Segment:
起始:CTU 承运人:CA 终到:BJS 至此航段使用里程制运价 运价：5914.06
总运价:NUC 13222.59 ROE:6.841320 CNY 140.00CN
CNY 28.00YQ
CNY 750.00YR

02 $ CRT+CIF 0058698.00 CNY INCL TAX
03 $ YRT+YIF 0037878.00 CNY INCL TAX
04 $ EKGV+YIF 0025638.00 CNY INCL TAX
05 $ DOWCN+C 0024888.00 CNY INCL TAX
06 $ QPR6MA+YI 0022398.00 CNY INCL TAX
07 $ EKGVO+Y 0009718.00 CNY INCL TAX
08 $ KKAPOCN1+ 0007038.00 CNY INCL TAX
**4,5,6,7,8*FLIGHT RESTRICTIONS APPLY*

**5,6,7,8*TKT STOCK RESTR*

**ATTN CHANGE TO NO STOPOVER MAY VARY FARE*

**ATTN PRICED ON 16JUN09*1605*

TIPS：

请求中的出票航空公司参数 validateCarrier 设置与否，会关系到查询结果的税项种类。
FSP 接口查询结果不会包括 XT 税项，而是返回 XT 税的明细税项

出错信息提示:

XSFSPCityException	缺少姓名组
XSFSPDateException	航段不能订取
XSFSPFormatException	航段组不连续
XSFSPRouteCodeException	

10. 信箱处理

实现类：com.travelsky.ibe.client.Queue
该类实现了处理 Q 的功能，包括发送 Q 到对方信箱，从本 Office 的信箱中收取 Q。目前，只能处理 office Q，不能处理工作号的 Q。

10.1. 察看 Office Q 的状态。

➢ eTerm 上察看 Office Q 的状态操作如下，非常简单的指令:

```
QT:
```

QT：显示信箱状态指令，可对应 getQueueContent()方法。

主机返回该 Office 的 Q 状态：

```
                    QT PEK099
GQ 3086 9999    RP 5007 9999    SE 0367 9999    CA 0000 9999
CG 0000 3360    CN 0079 9999    FG 0001 9999    FN 0039 9999
FQ 0028 1000    HT 0000 9999    KK 2501 9999    MD 0023 9999
MT 0023 9999    NA 1071 9999    RC 0000 0200    RE 0000 9999
RQ 0172 9999    SR 0593 9999    TC 0184 9999    TL 2201 9999
UG 0546 1000    UN 0122 9999    WC 0000 0500    SC 3004 9999
AV 0064 9999    XA 1779 9999    RJ 0001 9999    NC 0000 1000
AF 0000 9999    GN 0000 1000    BS 2433 9999    RL 0001 9999
RT 2900 9999    PB 0464 9999    IB 0000 9999    DS 3111 9999
HE 0000 9999    QX 0000 1155    HD 5744 9999    NB 0125 9999
AB 1263 2000    BE 5557 9999    BF 0000 5000    EE 0004 9999
```

每个 Office 都有一个或多个信箱，对于信箱中的内容，营业员应当尽快阅读并处理。
如上，PEK099 这个 Office 的 GQ 信箱中有 3086 条消息，GQ 信箱的容量上限为 9999 条。每个信箱名的含义可以参见订座手册，一般，如果某个 office 没有某个名字的信息，而其他 office 将 Q 发到该 Office，则该消息会进入这个 Office 的 GQ 信息。一般情况下，任意一个 Office 至少有 GQ 信箱。

➢ IBE 实现 Office QUEUE 状态获取

```
/*生成 Q 处理对象*/
QUEUE q = new QUEUE();

QueueContent content = q.getQueueContent();
```

获取状态之后，可以按顺序查看所有信箱的状态，程序输出附后：

```
/*展示 Office 号*/
System.out.println(content.getOfficeCode());
/*按顺序展示信箱的状态*/
for (int i = 0; i < content.getQueueBoxCount(); i++){
        QueueBoxItem item = content.getQueueBoxes(i);
        /* 信箱名*/
        System.out.print(item.getBoxName());
        System.out.print("        ");
        /*   未处理 QUEUE 数/信箱容量*/
        System.out.println(item.getNewQueue()+"/"+item.getCapacity());
}
```

```
PEK099
GQ      3086/9999
RP      5007/9999
SE      367/9999
CA      0/9999
CG      0/3360
CN      79/9999
FG      1/9999
FN      39/9999
FQ      28/1000
HT      0/9999
KK      2501/9999
MD      23/9999
MT      23/9999
NA      1071/9999
RC      0/200
RE      0/9999
RQ      172/9999
SR      593/9999
TC      184/9999
TL      2201/9999
UG      546/1000
UN      122/9999
WC      0/500
```

```
SC      3004/9999
AV      64/9999
XA      1779/9999
RJ      1/9999
NC      0/1000
AF      0/9999
GN      0/1000
BS      2433/9999
RL      1/9999
RT      2900/9999
PB      464/9999
IB      0/9999
DS      3111/9999
HE      0/9999
QX      0/1155
HD      5744/9999
NB      125/9999
AB      1263/2000
BE      5557/9999
BF      0/5000
EE      4/9999
```

也可以直接获取某信箱的状态，方便不同的专门信箱处理程序调用：

```
/*也可以按信箱名直接获取特定信箱状态*/
QueueBoxItem item = content.getQueueBoxes("SC");

/* 信箱名*/
System.out.print(item.getBoxName());
System.out.print("        ");
/*   未处理 QUEUE 数/信箱容量*/
System.out.println(item.getNewQueue()+"/"+item.getCapacity());
```

```
SC      3004/9999
```

10.2. 发送 QUEUE 到邮箱

营业员可以向任意一个 OFFICE 的邮箱 QUEUE 发送消息，格式有两种，一是发送 PNR 内容，

二是自写文本。
对方 OFFICE 接收消息时，如果发送方指定的邮箱在对方 office 中存在，将保存在该邮箱待处理消息的末尾，否则，将被送到对方的 GQ 邮箱的待处理消息末尾。

➢ eTerm 发送信息过程

eTerm 中运用主机指令发送自由文本消息过程如下：
第一行为指令，在第二行开始写消息，可以多行输入，然后在最后一行输入后 F12 提交，第一栏为指定时间发送，第二栏不指定时间，即时发送。

```
QE:GQ/CMS001/31JUL/1242
THIS is A TEST MESSAGE
                                                        ①
```

```
QE:GQ/CMS001
THIS is A TEST MESSAGE
```

eTerm 中运用主机指令发送 PNR 内容消息：

```
QE:GQ/CMS001/X042W                                      ②
```

① 发送自由文本消息：对应方法为 sendMessage(String mailbox, String officeCode,String message, Date dispTime, boolean priority)方法。
mailbox: 对方邮箱名
officeCode: 对方 office 代码
message: 要发送的消息
mailbox，officeCode，message 为必须录入项。表明发送的消息接收目标和内容。
dispTime: 设置此值非 null 则表示定时发送的时间，如即时发送可设为 null。
priority: 设置此值表示优先发送

② 发送 PNR:对应方法为 sendPNR(String mailbox, String officeCode, String pnr, Date dispTime, boolean priority)
mailbox: 对方邮箱名
officeCode: 对方 office 代码
pnr: 要发送的 pnr 编码
mailbox，officeCode，message 为必须录入项。表明发送的消息接收目标和内容。
dispTime: 设置此值非 null 则表示定时发送的时间，如即时发送可设为 null。
priority: 设置此值表示优先发送

如果正常发送，则上述两个方法都应当是返回 OK，否则会抛出异常，异常列表将附在本章末尾。发送结果可参看下一节：读取 QUEUE 中的待处理消息

➢ IBE 实现 Q 发送

```
        /*生成Q处理对象*/
```

```
QUEUE q = new QUEUE();
/*即时发送自写文本到CMS001 的GQ*/
String sendResult1 = q.sendMessage("GQ","CMS001","THIS is A TEST MESSAGE");
/*发送自写文本到CMS001 的GQ，定时2009 年7 月31 日12 时42 分发送*/
Calendar cal = Calendar.getInstance();
cal.set(2009,6,31,12,42,0);
String sendResult2 =q.sendMessage("GQ","CMS001","THIS is A TEST MESSAGE",cal.getTime());
/*发送自写文本到CMS001 的GQ，即时发送，优先*/
String sendResult3 =q.sendMessage("GQ","CMS001","THIS is A TEST MESSAGE",null,true);

 /*即时发送PNR：X042W 的内容到CMS001 的GQ*/
String sendResult4 =q.sendPNR("GQ","CMS001","x042w");
/*发送PNR：X042W 的内容到CMS001 的GQ 定时2009 年7 月31 日12 时42 分发送*/
String sendResult5 =q.sendPNR("GQ","CMS001","x042w",cal.getTime());
/*发送PNR：X042W 的内容到CMS001 的GQ 定时2009 年7 月31 日12 时42 分发送,优先*/
String sendResult6 =q.sendPNR("GQ","CMS001","x042w", cal.getTime(),true);
```

10.3. 读取并处理 Q

接收到 Q 之后，需要尽快从信箱中读出，根据情况，可以做以下处理 ：

1、处理完，将消息从队列中清楚

2、未处理，或有其他原因稍后处理，则可以把该指令放到待处理消息队列末尾，下次再处理。

➢ eTerm 操作过程

eTerm 中运用主机指令读取 Q 指令及主机响应如下

```
QS:GQ
CMS001 GENERAL MSG   ( 0002 ) ( 0002 )
THIS is A TEST MESSAGE
FROM AGENT 8881  PID45562  CMS001  0603  31/07/09
QDE

QS:GQ
CMS001 GENERAL MSG   ( 0002 ) ( 0002 )
 1.张林军 X042W
 2.  ZH9702 U   TU28JUL09WUXSZX RR1   1620 1820         E --B
 3.CZX/T CZX/T 0510-82723223/WUXI OVERSEA TOUR TICKET CO.,LTD/SHI HUA ABCDEFG
 4.REM 0715 1017 CZX113-5
 5.88553333
 6.TL/1800/27JUL09/CZX113
```

```
 7.FC/A/WUX ZH SZX 520.00YTJ40 CNY520.00END
 8.SSR FOID
 9.OSI 1E TKNO/479-2432470637 1ZHANGLINJUN
10.RMK CA/LEWH9
11.RMK AUTOMATIC FARE QUOTE
12.FN/A/FCNY520.00/SCNY520.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY570.
00
13.EI/BUDEQIANZHUANBUDEZIYUANTUIPIAO不得签转不得自愿退票
14.FP/CASH,CNY
15.CZX113
QNE
```

QS：GQ　打开 GQ 信箱，主机会自动显示当前 GQ 中的第一条消息。可见该消息为自由文本消息。我们的程序默认只取一条消息，因此在读取完之后会退出信箱

QDE　表示放弃处理当前消息并退出，当前消息会移到队列末尾

QS：GQ　再次打开 GQ 信箱，主机会自动显示当前 GQ 中的第一条消息。可见此时该消息为 PNR 内容。我们的程序默认只取一条消息，因此在读取完之后会退出信箱

QNE　表示已经处理当前消息，将其从信箱删除之后并退出

读取 Q 的操作对应 getQueue(String boxName, boolean release, boolean decode)方法。该指令每次从制定的 Q 中获取第一条消息然后退出，可以选择退出时从信箱中删除或者不删除获得的消息。

boxName:　信箱名

release:　读取消息后是否释放，如果调用 Q 读取的程序只是为了获取消息但并不处理，另有处理机制，则可以设置为 false。如程序读出后有专人处理并记录，则可以设置为 true。

decode:　是否将消息的内容尝试当成 PNR 显示来解析。如果设置为 true，则会将每条消息都当成文本读取之后再尝试给出结构化的 PNR 数据，如果设置为 false，则所有消息，包括自由文本消息和 PNR 消息都按照文本格式返回给调用客户端。

- IBE 实现成人缺口程 PNR 预定

第一句 getQueue 获取 Q 的操作选择为从 GQ 中读取一条消息，并在读取之后不删除，也不尝试按 PNR 解析。

第二句 getQueue 获取 Q 的操作选择为从 GQ 中读取一条消息并尝试按 PNR 解析，读取之后删除。从输出内容可以看到与 RT 指令一样的 PNR 解析结果。

```
    /*生成 Q 处理对象*/
QUEUE q = new QUEUE();

System.out.println(q.getQueue("GQ",false));

System.out.println(q.getQueue("GQ",true,true));
```

CMS001 GENERAL MSG (0002) (0002 (0)
THIS is A TEST MESSAGE
FROM AGENT 8881 PID45562 CMS001 0603 31/07/09

PNR Code: null
Deleted Flag: false
Decode Result:
No Decode

CMS001 GENERAL MSG (0002) (0002 (0)
1.张林军 X042W
2. ZH9702 U TU28JUL09WUXSZX RR1 1620 1820 E --B
3.CZX/T CZX/T 0510-00000000/ ABCDEFG
4.REM 000000000 CZX
5.00000000
6.TL/1800/27JUL09/CZX000
7.FC/A/WUX ZH SZX 520.00YTJ40 CNY520.00END
8.SSR FOID
9.OSI 1E TKNO/479-0000000000 1ZHANGLINJUN
10.RMK CA/LEWH9
11.RMK AUTOMATIC FARE QUOTE +
12.FN/A/FCNY520.00/SCNY520.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY570.00 -
13.EI/BUDEQIANZHUANBUDEZIYUANTUIPIAO 不得签转不得自愿退票
14.FP/CASH,CNY
15.CZX000

PNR Code: X042W
Deleted Flag: false
Decode Result:
PNRNO:X042W

团体票标识:散客票

姓名组
1.1.张林军 成人 张林军

航段组
1. 2. ZH9702 U WUXSZX RR1 Tue Jul 28 16:20:00 CST 2009 Tue Jul 28 18:20:00 CST 2009 电子票 离港航站楼:-- 到港航站楼:B 普通航段
ZH9702 U TU28JUL09WUXSZX RR1 1620 1820 E --B

联系组
1.3.CZX CZX/T CZX/T 0510-00000000/ ABCDEFG

2.4. REM 000000000 CZX
3.5. 88553333

开账地址组（BA）

出票组
1.6.TL Mon Jul 27 18:00:00 CST 2009 CZX000　　备注:TL/1800/27JUL09/CZX000
票号组

备注组
1.10.CA/LEWH9
2.11.AUTOMATIC FARE QUOTE

特别服务组
1.8.SSR　FOID　null　　0　null null

其他服务组
1.9.1E TKNO/479-0000000000 1ZHANGLINJUN

婴儿组

其他
1.7.FC/A/WUX ZH SZX 520.00YTJ40 CNY520.00END
2.12.FN/A/FCNY520.00/SCNY520.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY570.00
3.13.EI/BUDEQIANZHUANBUDEZIYUANTUIPIAO 不得签转不得自愿退票
4.14.FP/CASH,CNY

FN 组
1.原文：FN/A/FCNY520.00/SCNY520.00/C3.00/XCNY50.00/TCNY50.00CN/TEXEMPTYQ/ACNY570.00
票面价：CNY520.00
实收：CNY520.00
代理费率：3.00
附加费用之和：CNY50.00
税项：　　　CN:CNY50.00　　　YQ:EXEMPT
总价：CNY570.00

FC 组
1.原文：WUX ZH SZX 520.00YTJ40 CNY520.00END
Segment:
*　　起始:WUX　承运人:ZH　终到:SZX　运价：520.00　运价基础：YTJ40*
总运价:CNY　　520.00

FP 组
1.14.FP CASH,CNY　　原文:FP/CASH,CNY

OI 组

TC 组

EI 组

1.13.EI EI/BUDEQIANZHUANBUDEZIYUANTUIPIAO 不得签转不得自愿退票

责任组

1.15. CZX000

出错信息提示

QIllegalException	由于数字代码错误导致非法操作
QInvalidFormatException	输入格式错误导致操作被拒绝
QNoQueueException	需要读某 OFFICE 的某信箱,但该 office 此信箱不可用
QQueueEmptyException	所访问的信箱已空
QRecordNotExistException	发送 PNR 格式的消息，但 PNR 编号不存在